Michal Kopček

Optimal Pilot Bus Selection for the Secondary Voltage Control Using Parallelism

Scientific Monographs in Automation and Computer Science

Edited by
Prof. Dr. Peter Husar (Ilmenau University of Technology) and
Dr. Kvetoslava Resetova (Slovak University of Technology in Bratislava)

Vol. 10

OPTIMAL PILOT BUS SELECTION FOR THE SECONDARY VOLTAGE CONTROL USING PARALLELISM

Michal Kopček

Universitätsverlag Ilmenau
2013

Impressum

Bibliographic information of the German National Library
The German National Library lists this publication in the German national bibliography, with detailed bibliographic information on the Internet at http://dnb.d-nb.de.

Author’s acknowledgement to Gabriela Chmelíková for translation.

This scientific monograph originated from the author's dissertation thesis defended at the Slovak University of Technology in Bratislava, Faculty of Materials Science and Technology in Trnava.

Reviewers:
Prof. Ing. Juraj Spalek, PhD.
Prof. Ing. Mária Franeková, PhD.
Prof. Ing. Alojz Mészáros, PhD.

Author’s contact address:
Ing. Michal Kopček, PhD.
Slovak University of Technology in Bratislava
Faculty of Materials Science and Technology in Trnava

Technische Universität Ilmenau / University Library
Universitätsverlag Ilmenau
Postfach 10 05 65
D-98684 Ilmenau (Germany)
http://www.tu-ilmenau.de/universitaetsverlag

Production and delivery
Verlagshaus Monsenstein und Vannerdat OHG
Am Hawerkamp 31
D-48155 Münster (Germany)
http://www.mv-verlag.de

ISSN 2193-6439 (Print)
ISBN 978-3-86360-077-8 (Print)
URN urn:nbn:de:gbv:ilm1-2013100075

Titelfoto: photocase.com

Abstract

The scientific monograph submitted deals with the subject of power system set and its control on the secondary level. The power system can be characterized due to its size, complexity and load unknown in advance, in addition to time variability. Since large electric power is transferred, even small changes in setting can frequently represent significant economic savings. Regarding the power system properties as well as the basics of secondary control it is obvious that the optimal selection of pilot bus plays an important role in the quality of control and the related economic impact.

The monograph focuses on the methods of optimal selection of the pilot buses for the dispatch control of bus voltage in a complex power system. The aim is to modify them so that it is possible to investigate the issue via using the parallelism principle and grid computer structure aimed at multiple shortening of the computing time.

In Chapter 1 the author describes the power system, defines it, and analyzes its control as well as its physical nature. In Chapter 2 he deals with the theoretical background of the pilot bus selection, e.g. with purpose functions and individual methods of optimal pilot bus selection. In Chapter 3 the author precisely defines the subject investigated and the aims. Chapter 4 is devoted to analysis of the possibilities for utilizing the parallelism principle and brings the design and the prototype implementation of the system architecture of optimal pilot bus selection of the bus voltage dispatch control for a complex power system, to enable operation in real time. Chapter 5 experimentally verifies the architecture designed.

Key words

optimal pilot bus selection, grid computing, parallelism, genetic algorithms

Scientific contribution

The main benefits of the monograph submitted can be summarized as follows:

- Design of original system architecture of optimal pilot buses selection for the dispatch control of complex power system in real time.
- Application of an original approach to the solution of optimal pilot buses selection based on the principle of parallelism and possibility of the use of algorithm grid structure.
- Implementation of original prototype software declaring the functionality of the architecture designed via the experiments carried out.
- Experimental verification of method correctness verification and considerable shortening of computing time of the method.
- Original application of parallel genetic algorithms for the solution of the principal subject of the monograph.
- Experimental verification of the originally designed architecture capability to operate in the environment of grid computer structure.

The designed original system architecture for the optimal pilot buses selection for the dispatch voltage buses control of the complex power system capable to operate in real time is the final result; and regarding the partial objectives determined, the main aim of the monograph submitted was met.

LIST OF ABBREVIATIONS

ES	– Power system
PES	– Interconnected power system
UCTE	– Union for the Co-ordination of Transmission of Electricity
ENEL	– Ente Nazionale per l'Energia Eletrica
USA	– United States of America
SG	– Synchronous Generator
GA	– Genetic Algorithm
BOINC	– Berkeley Open Infrastructure for Network Computing
LAN	– Local Area Network
MODES	– software package for power system simulation
UST	– part of MODES package for calculation of ES stable state operation
UNIX	– operation system
MS Windows XP	– operation system by Microsoft Company
OKIS	– Division of Communication and Information Systems
PC	– personal computer

LIST OF SYMBOLS

f_{Ukv}	– minimum of squared voltage deviation
f_{Qp}	– minimum of transferred reactive power
$f_{\Delta Ur}$	– minimum of maximum relative voltage deflection
f_K	– multi-criterion purpose function
σ	– threshold value of the sensitivity coefficient
$C_n(k)$	– number of possible candidates for pilot buses
n	– dimension of searched nth-tad (number of searched pilot buses for the ES)
k	– size of candidates set (number of candidates for one pilot bus)

INTRODUCTION

Electric power is an inseparable part of our everyday life, and almost no one can operate a company, a factory or even a household without the facilities wholly dependant on the electric power source. To utilize these facilities carelessly, the electric power transported to the location of its consumption has to meet the prescribed values in terms of voltage and frequency. This transmission is mediated by the power system.

A power system is characteristic by its size, complexity and load which is unknown in advance and variable in time. It is obvious, that the power system has to transfer large electric powers, and even small changes in setting can frequently represent significant economic savings. Therefore, the subject is worthwhile to investigate and research further.

Voltage control, which is one of the most important qualitative factors, is dealt with on the secondary power system control level. The principle of secondary voltage control is, on the side of consumers, in keeping the constant voltage value via the control of the synchronous generators´ clamping voltage. The control of the clamping voltage runs on the basis of voltage level monitoring in several locations selected in the power system called pilot buses. Regarding the power system properties, it is obvious that the optimal selection of pilot buses plays a key role in the issue of control quality and related economic impact.

The subject of the optimal pilot buses selection is comprehensively described in contributions 21, 31, in which the authors developed several algorithms for the purpose. The scientific monograph submitted deals with the methods of the optimal pilot buses selection of the bus voltage dispatch control for a complex power system to be able to modify them, so that the real issue can be handled via using the principle of parallelism and grid computer structure for the multiple computing time shortening.

The monograph comprises five principal chapters and annexes complementing the overall picture of the subject. In Chapter 1 the author describes the power system, defines it, and analyzes its control as well as its physical nature. The analytical part of the monograph is included in Chapter 2 in which he deals with the theoretical background of the pilot bus selection, e.g. the purpose functions and individual methods of optimal pilot bus selection. In Chapter 3 the author precisely defines the subject investigated and the aims. The penultimate Chapter is devoted to the analysis of the possibilities for utilizing the parallelism principle and brings the design and the prototype implementation of the system architecture in the system of optimal pilot bus selection of the bus voltage dispatch control for a complex power system, to be able to operate in real time. In the last chapter (Chapter 5) the author experimentally verifies the architecture designed from the point of the evaluation of the predictions correctness and within the context of its physical implementation in the dispatch control in real time conditions.

1. POWER SYSTEM AND ITS CONTROL

The power economy of any country can be defined as a large system characterized as a central and homogenously controlled set of interconnected and mutually influencing power devices. It includes the acquisition of all natural energetic resources, thermal energy and electric power, their transmission, distribution and change, including the change in consumer devices to applicable kinds of energy directly utilized in all technological, economic-communal and household consumer processes 20.

A power system is a central and homogenously controlled set of interconnected and mutually influencing power devices including the operating teams servicing the power system in the company by a specific fuel or power.

The most important part of the energetic system is the power system. As aforementioned, it is a central and homogenously controlled set of interconnected and mutually influencing power plants, power control devices and electric appliances. In other words – the power system (ES) as a subsystem of the energetic system deals with the generation, transmission, distribution, change and use of the electric power.

1.1 Characteristics of the power system

The power system can be characterized by:

- size – ES covers a large location – territorial size;
- complexity – feedback among the individual ES parts;
- load random character – the ES load size unknown in advance.

In the complex ES operation, as in the control object, many troubles can occur, and they must be handled as a whole, as they can occur in the field of electric power generation, distribution and consumption. In addition, we have to consider mutual relations and the use of the most modern control means such as automates, control computers, microprocessor systems, etc.

There are several requirements for the ES to be met:

- reliability of the electric energy supply. The ES decomposition results in large damages as well as in other troubles in ES, e.g. interruption of international cooperation; activities of relay protections and system automation, etc.;
- economy of operation – it is mainly the optimal ES operation (minimum costs for ES operation);
- following the qualitative factors, particularly the frequency in the system and individual buses voltage in prescribed limits.

1.2 ES specifications from the point of control

ES can be distinguished from other systems by the following specifications:

- Generation and consumption of electric power takes place at the same time. In practice so far, there are no accumulators of alternative electric power.
- The speed of transmission actions run when compared to the operational manipulations. The wave, electromagnetic and electromechanic transmission phenomena run practically from

milliseconds to several seconds which requires the use of special devices in the control.

- Constant ES development is caused by the connection of ES and all other society branches.

Science and technology development in recent decades has brought new methods and means of information and control processing. This has reflected also in the control of complex ES which are nowadays controlled using the latest means of information collection, transmission and processing as well as the utilization of artificial intelligence elements (expert systems, artificial neural networks, fuzzy logic, genetic algorithms, etc.)

1.3 Interconnected power systems

International cooperation is applied for the purposes of mutual electric power exchange, in order to increase the system operation stability, especially for the solutions of malfunctions and emergencies.

ES can be operated:

1. independently;
2. in the connection with other ES.

In contrast to an independent ES, parallel cooperation of the interconnected power system (PES) has unambiguous advantages including:

- reliability increased in terms of electric power supply via mutual cooperation;
- electric power supply quality improvement due to more consistent frequency and related voltage;

- a decrease of reserve powers need;
- the possibility of utilizing the inter-system effect arising from the difference of daily load diagrams;
- the possibility to install blocks with higher unit performances and better operational properties in the individual ES;
- a decrease of electric power losses in networks. Through controlled parallel cooperation it is possible to achieve a better division of powers flow in transmission networks, and thus shorten the distances of transmissions;
- fuel savings – the possibility of utilizing the economic load division in PES.

Nevertheless, the interconnected system brings new issues for the control system, e.g.:

- the issue of control balances of transferred powers and energy according to long-term economic contracts;
- the issue of voltage control in system boundary points;
- the need for international coordination of maintenance in generation and distribution devices.

The interconnection of systems can be executed in one defined point via:

- galvanic connection by the parallel cooperation of systems;
- unidirectional control or unidirectional connection by the different frequency of systems;
- specification of one system sources operating to the other system (island operation);
- specification of load islands of one system to the other one.

Slovak ES is a part of UCTE interconnected system from 17 May 2001, when the country became its legitimate member.

1.4 Classification of buses in ES

The solution to stable operation in ES is one of the subjects to be dealt within ES operation management. In the calculation of stable ES operation we determine the:

- division of active and reactive powers;
- losses of powers and energy;
- voltage levels in individual network buses;
- loading of individual network parts;
- cross-section of conductors and cables;
- maximum transferred power regarding stability.

The calculations of stable operations in complex ES have their specifications including the:

- necessity to use the iterative calculation method regarding the mathematical model which is in general described by the system of non-linear algebraic equations;
- necessity to consider technical possibilities in terms of sizes of voltage, maximum currents (from the heat dimensioning conditions), Qmax – which is at the disposal, maximum transferred power, etc.;
- complexity – high number of elements, buses and branches.

The stable operation is unambiguously determined, if we know the values in the bus including the:

- absolute value of voltage U;
- voltage angle δ;
- active power P;
- reactive power Q.

Two values in the bus are usually given and the others can be obtained by the solution to the stable operation in ES. According to which of the bus values are given we can classify the buses as follows:

1. Class (U, δ) – balance bus. The bus is usually marked in the substitute scheme as the first one and by the solution to the stable operation we get the active and reactive powers. The task of the bus is to balance the powers considering the active and reactive losses in networks. Therefore, we usually select one of the connectors as the balance bus comprising sufficiently large sources of powers.
2. Class (P, Q) – connecting or consuming buses, in which the active and reactive powers are given. The powers of sources are distinguished by a sign. By implementing the solution to the stable operation we can find the voltage and its angle.
3. Class (U, P) – control or compensation buses. In these buses the absolute value of voltage and the active power are given. By implementing the solution we obtain the voltage angle and the reactive power value (supplied or consumed) needed for keeping the voltage given.

In the practical calculations of stable ES operation we usually assign:

- one balance bus (U, δ);
- *k* of buses (P, Q);
- (*n-k-1*) of buses (U, P).

1.5 Power system control

The subject of the power system control is a very complex process. It is the complex control system whose individual components are arranged over a large area. Its activity is influenced by a number of factors, whereas some of them are not known in the classical control systems at all 17.

1.5.1 Electric power quality control

The quality of the electric power is characterized mainly by the frequency in ES and by the voltage in system buses. The deflections from these factors are strictly determined by the standard; therefore, it is necessary to ensure their control, so that they are kept within the allowed limits.

In the stable operation of the power system we assume the balance between the generation and consumption of the power at any moment in time, i.e. the balance of active and reactive powers. Any disturbance of this balance causes the change of frequency, or voltage in the power system. This change remains until the power balance is repeatedly balanced. The balance is caused in the controlled as well as in the non-controlled systems, obviously in a different way.

The balance of powers in ES stands only for certain frequency values and voltages in the system. Through their changes, the result is a change of generated or consumed powers. It stands also vice versa, i.e. by the change of the load or by the change of generated powers in the system it results in a change of the frequency and the voltage. The frequency in ES is characterized by the balance of the active powers in the whole ES and the voltage is characterized by the balance of powers only in the given area. Therefore, we can say that the frequency is a global operational parameter and the voltage is a local operational parameter of ES.

1.5.2 Control of voltage in ES

The subject of the voltage values control and the changes of the reactive power in ES can be described as a massive control issue with a multilevel and multiobject control structure 7, 8, 9, 10, 16. Due to the simplification of the control implementation the cascade control is normally used. The control levels are divided into the hierarchic levels with a pyramidal structure.

From the perspective of response time speed and the hierarchy, the voltage control can be divided into primary, secondary and tertiary controls.

The voltage value in the boundary power lines has to harmonize the neighboring system operators so that the flows of the reactive powers can be managed. If the voltage deflections are a regular issue in the neighboring systems, it is necessary to keep the voltage within the normal voltage range by the compensation possibilities of the partners. The employment of these compensation devices has to be agreed by the related members.

The optimization calculation is carried out on the Slovak Electricity Transmission System, plc (SEPS). Regarding the measurement and signalization in real time, the optimal parameter values of the voltage control can be determined, i.e. the recommended size of the voltage in the pilot buses, voltage transmission of transformers, the employment of compensation means, etc. for the pilot buses. The implementation of the active impacts from the calculated values is ensured via the secondary voltage control.

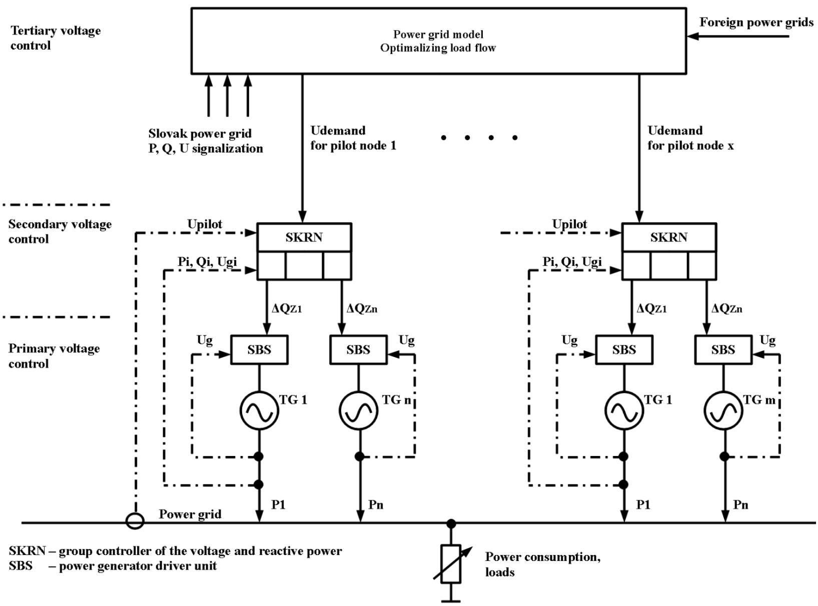


Fig. 1 *Principle scheme of voltage control*

In this structure (Fig. 1) some controllers do not have the direct access to ES. Therefore, some controllers regarding the information from the controllers shown at the higher control level also control the controllers shown at the lower. The controllers shown at the secondary voltage control level are a typical example of the controllers without the direct access to ES.

Primary voltage control

Primary voltage control 7, 8, 9, 10, 16 is executed by the voltage controllers in the exciting alternators circuits. On this level the control elements try to balance the severe and accidental voltage deflections by the continuous maintenance of their outcome variables close to the required reference values. For the calculations of the control signals only local information is used. We assume a quick response; the time constant is in the range of a few seconds. The operational range of excitation is given by several limitations: allowed flow of the rotor, stator, heating, etc. The primary voltage controllers usually do not control the boundary limits of the terminal voltage and own consumption.

Secondary voltage control

Controllers of the secondary voltage control level 7, 8, 9, 10, 16 are responsible for the slow and large voltage deflections, such as those caused by the load development lasting for an hour. The time constant is in the range of minutes. This level utilizes the regional information to update the reference values of the primary voltage control level controllers to maintain the voltage on the pilot buses in optimal values.

The role of the secondary voltage control 31, 21 is in the setup of the primary controllers' reference values in a coordinated way with the achievement of the required operation of the whole system. The secondary voltage control determines the impacts of control devices based on the reference voltage levels set on the specific load buses determined as the pilot buses. The secondary voltage control automatically coordinates the reactive power sources in order to achieve a suitable voltage profile and to maintain the satisfactory reactive powers reserves. Through the use of the secondary voltage control it is possible to maintain the suitable voltage profile in the whole ES despite the hour load development and the topologic changes.

The secondary voltage control has to respect all the limitations given by the alternator, block transformer and own consumption and must not cause unallowed operational parameters.

Tertiary voltage control

To calculate the optimal voltage values on the pilot buses we utilized information from the whole system to achieve the economic and safe activity of the power network. This is achieved mainly by the solution (automatic or manual) to the large optimizing issue, as the optimal power flow is aimed at the minimization of the active power losses, whereas the safety limitations are considered. The tertiary voltage control level is responsible for the coordination, i.e. all the network is considered. This means, that it is significantly slower than the secondary voltage control level, the time constant is in the range from 15 minutes to several hours 7, 8, 9, 10, 16.

1.5.3 Mathematical description of the secondary voltage control

The selection of the pilot buses represents the key factor for the correct activity of the secondary voltage control 7, 8, 9, 10, 16. Their selection is based either on simple rules or on linearized equations formulations of the load flow as frequently mentioned in various sources. Nevertheless, if the system operates close to the limit of the transmission capacity, the importance of the secondary voltage control is more obvious. In such a situation the use of the linearized model system is unsuitable and the set of the pilot buses selected via the linearized model can be far from the optimum. Therefore, it is necessary to use the model which does not include the non-linearities as well.

The secondary voltage control operates in an incremental way. Each control step changes the voltage value on the load buses by a quite small value. Due to this it is possible to use the linearized control model.

The used linearized model considers the flows of the active power in the system; however, it neglects the influence of changes of the active power on the voltage values. The model is based on the following equations of sensitivity:

$$\begin{bmatrix} K_{GG} & K_{GZ} \\ K_{ZG} & K_{ZZ} \end{bmatrix} \cdot \begin{bmatrix} \Delta \boldsymbol{v}_G \\ \Delta \boldsymbol{v}_Z \end{bmatrix} = \begin{bmatrix} \Delta \boldsymbol{q}_G \\ \Delta \boldsymbol{q}_Z \end{bmatrix}, \qquad [1]$$

where:

K_{GG}, K_{GZ}, K_{ZG} and K_{ZZ} are sensitivity matrices:

K_{GG} – is the own sensitivity matrix of generator buses;

K_{ZZ} – is the own sensitivity matrix of the load buses;

K_{GZ}, K_{ZG} – the mutual sensitivity matrix of the load and generatoric buses;

$\Delta \boldsymbol{v}_G$ a $\Delta \boldsymbol{v}_Z$ – vectors of voltage values changes on individual generatoric and load buses;

$\Delta \boldsymbol{q}_G$ a $\Delta \boldsymbol{q}_Z$ – vectors of changes of the supplied reactive power on the individual generator and load buses;

Vector $\Delta \boldsymbol{q}_Z$ is the cause of the changes in the system, vector $\Delta \boldsymbol{v}_G$ is the control vector. It further stands that:

$$\Delta \boldsymbol{v}_z = M \Delta \boldsymbol{q}_Z + B \Delta \boldsymbol{v}_G , \qquad [2]$$

where

$M = K_{ZZ}^{-1}$

$B = -K_{ZZ}^{-1} K_{ZG}$.

The previous equation describes the change of the voltage on the load buses caused by the load deflections (the voltage changes after unpredictable events originated prior to the secondary voltage control impact) and the effect of the control generators. The expression considers also the cases of more concurrent unpredictable events when the related members $\Delta \boldsymbol{q}_Z$ are not zero.

If it is possible to calculate the input control quantities automatically in real time, then the feedback control represents the solution. I will assume a linear feedback control law (related to the secondary voltage control) using the voltage deflections $M \Delta \boldsymbol{q}_Z$ (or only some members of this vector) as the own input and then generates the control impacts $\Delta \boldsymbol{v}_G$. If the number of generators n_G is smaller than the number of load buses n_Z, it is not possible to find the control $\Delta \boldsymbol{v}_G$ leading to zero voltage deflections $\Delta \boldsymbol{v}_Z$ after the malfunction.

The measurement achievability in all load buses is another consequence important for the implementation as well as for the theoretical considerations. If only some $M\Delta \boldsymbol{q}_Z$ vector members are available for the secondary voltage control, the structure of the pilot buses is necessary. This simplifies the controllers' structure; however, it decreases the efficiency. The design of the controller operating with reduced information and acceptable performance is then the optimization task.

The load bus with numerous large members (absolute values) in the related line of *M* matrix is a suitable candidate for the pilot bus. This procedure modification was used in Italy. However, if we want to consider the location of the generators regarding the load (given by *B* matrix), it also influences the pilot buses selection and therefore, the more exact optimization method is needed.

The controllers will have the voltage values on the pilot buses available, the equation of observability is as follows:

$$\Delta V_p = C\Delta \boldsymbol{v}_Z \, , \tag{3}$$

where

C=[c] – matrix of zeros and ones with dimensions $n_p \times n_z$ defined:

$c_{ij} = 1$, if is *j*-th bus by *i*-th pilot bus and 0 in other case;

n_z is the number of load buses;

n_p is the number of load buses determined to be the pilot buses.

The secondary voltage control tries to minimize the voltage deflections on the load buses by using the outcome information available, i.e. the voltage values on the pilot buses. The control generators compensate the load malfunctions by maintaining the stable state of the voltage values on the pilot buses:

$$\Delta V_p = CB\Delta \boldsymbol{v}_G + CM\Delta \boldsymbol{q}_Z = 0 . \qquad [4]$$

This can be caused due to various reasons, since the number of control generators (control variables) is larger than the number of pilot buses (controlled variables). The minimization of the control generators impacts via the following relationship:

$$\sum_{i=1}^{n_G} \Delta V_{Gi}^2 \to \min , \qquad [5]$$

where

n_G – the number of control generators.

The previous equations show the control law defined by the expression:

$$\Delta V_G = -FCM\Delta Q_Z , \qquad [6]$$

where

$\mathrm{F} = (\mathrm{CB})^{\mathrm{T}} (\mathrm{CBB}^{\mathrm{T}} \mathrm{C})^{-1}$.

This expression includes reduced information. The matrix C selects n_P of lines from the matrix M (i.e. $M\Delta Q_Z$ vector members), which corresponds with the pilot buses. The matrix F is the matrix of gains to be determined. Then, after $n_Z \times n_Z$ implementation of the unit matrix I_Z I obtain the equation:

$$\Delta V_Z = M(I_Z - K_{ZG}FCM).\Delta \boldsymbol{q}_Z = fnc(F,C).\Delta \boldsymbol{q}_Z . \qquad [7]$$

It is obvious that the design of the secondary voltage control 20 comprises two sub-issues:

- Selection of pilot buses (matrix *C*);
- Selection of gains (matrix *F*) showing the voltage deflections on the pilot buses to the changes of reference income values of the generators.

The first issue is represented by the discrete optimization task with a very large set of possible solutions (all combinations of n_p z n_z buses) and the only way to ensure the optimization is to verify the whole set of possible solutions 18. Due to many dimensions of the task it is not possible to carry it out for the real size power systems, however there are several possible approaches which do not have to lead to the global optimum but provide good solutions.

1.5.4 Physical basics of the secondary voltage control

Fig. 2 illustrates a part of the power system, where SG is the synchronous generator, V_n are power lines and in the individual buses the consumption of the active and reactive power P_n, or Q_n are connected. The clamping voltage of U_g generator is regulated by the primary voltage control. U_{odb} voltages are on the consumption buses.

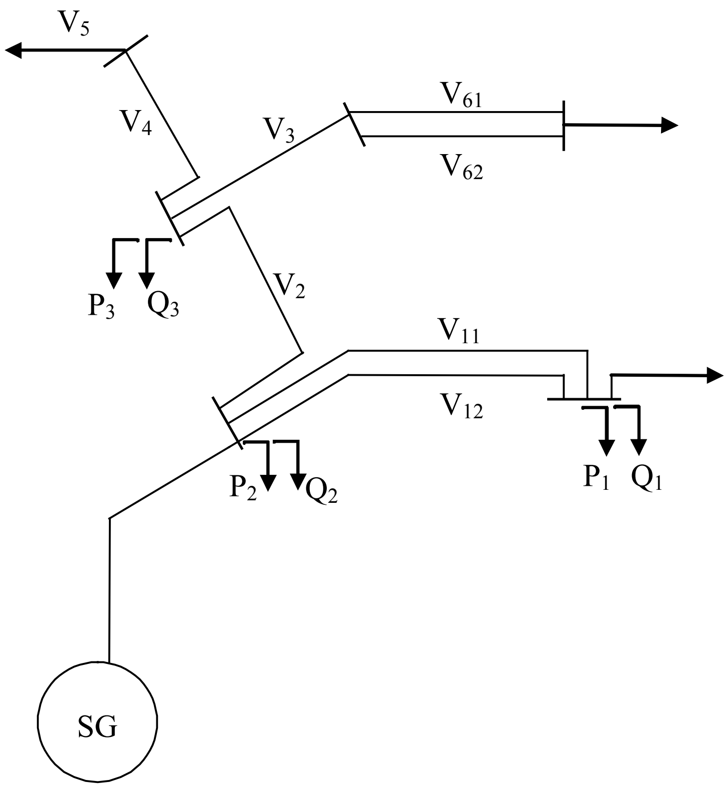


Fig. 2 *Scheme of the power system part*

Let us assume that the consumption of the idle generator increases. The physical consequence will be the voltage decrease in the given buses. The voltage U_g also decreases by a certain value; and subsequently it is modified to the required value by the primary voltage control. Nevertheless, as a result of losses in the lines, the voltage in the consumption buses does not achieve the original value, it will be lower. Alternatively, the feedback used in the primary voltage control is not suitable for the voltage control in the consumption buses. This basic drawback of the primary voltage control solves the feedback implementation from the consumption bus voltage. This method of control is called the secondary voltage control.

2. THEORETICAL OUTCOMES FOR PILOT BUSES SELECTION

The use of automatic voltage control in the power system controlled on several hierarchic levels continuously increases it significance17, 21, 31. Since the fully automated secondary voltage control is superior to the primary voltage control, the development of three-level control system becomes more significant. The secondary voltage control systems are developing in two directions. The first one is a decentralized secondary voltage control applied in systems comprising several mutually independent compact zones. The other direction is represented by the coordinated control of the secondary voltage, if the system is an inseparable complex:

- Decomposition of the areas of the secondary voltage control is used by the French power system network (Electricité de France). Experience shows that the implementation of the control system is complicated and time consuming (regional automation was accomplished between 1974 -1984).
- The power system network of Italy (ENEL) considers the secondary voltage control as a sequence of individual areas development with the pilot bus specified.

Both approaches aforementioned showed initial promising results and went through several modifications. The solution came out of heuristic approaches and in the next stage they tried to utilize the formal approach by using the systems theory. The effort was mainly seen in France, where the issue of coordination among the areas required the review of some methodology aspects.

Power systems in the USA are traditionally projected and operated so that the system support by the reactive power is available in each support function. The reactive power becomes a significant means of the power system daily operation in the normal and abnormal system modes.

2.1 Basic terminology

2.1.1 Selected purpose functions

The idea of pilot buses selection is based on the assumption that the selected set of pairs – pilot bus and generator bus – is of the kind by which the minimum value of the criterion function is achieved in the system random load by the reactive power under the condition that the pilot buses voltage values are maintained on the prescribed level. The task can be mathematically expressed as follows:

If the number of monitored load buses is n_Z and their voltages are $U_{Zi}, i = 1, 2, \ldots n_Z$. Then the number of pilot buses is $n_P < n_Z$ and their voltages are $U_{Pj}, j = 1, 2, \ldots n_P$. Through the change of the load in the system the voltages in the load buses change to the values of $\overline{U_{Zi}}, i = 1, 2, \ldots n_Z$. Due to the secondary voltage control the voltages in the pilot buses remain practically the same values. The criterion functions of the optimal pilot buses selection, their number as well as their position in ES are described further 21, 31.

Minimum value of the squared voltage deviation

From the point of a consumer it is decisive that through the selection of the candidate for the pilot bus within the secondary voltage control in ES, it is important that the voltage value in the consumption buses, under the condition that pilot bus voltage is maintained on the constant value, changes at a minimum by the change of the reactive power consumption in the load buses. This criterion is mathematically expressed as the addition of squared voltage deviations of the consumption buses 21, 31:

$$f_{Ukv} = \sum_{i=1}^{n_Z} \Delta U_i^{\,2} \ , \qquad [8]$$

where

ΔU_i – is the voltage deflection in the consumption bus by the increase of the reactive power consumption;

n_Z – number of load buses.

Minimum value of the supplied reactive power

The total sum of supplied generators´ reactive power through the increase of the reactive power consumption on the consumption buses is another candidate suitability criterion, particularly from the perspective of the power supplier. The value should be as low as possible, since the generation and transmission of the reactive power represents production losses for the supplier. Formally this function is expressed as the minimum of the supplied reactive power by all synchronous generators 21, 31:

$$f_Q = \sum_{i=1}^{n_G} Q_{dod\ i} \, , \qquad [9]$$

where

n_G – is the number of generator buses through the given minimum and maximum values of the reactive power supplied $Q_{dod\ i} \in \left\langle Q_{\min i}, Q_{\max i} \right\rangle$.

Minimum value of the maximum voltage deviation

The balance of voltage deflections of the consumption buses by the large consumption of reactive power can be a special criterion, i.e. this type of pilot bus selection guarantees the minimum value of the largest deviation. Mathematically taken, the function is defined as the minimum of the maximum voltage deviation value on the consumption bus 21, 31:

$$f_{\Delta U} = \max\left(\Delta U_{Z\ i}\right), \qquad [10]$$

where

$\Delta U_{Z\ i}$ – is the voltage deviation in the consumption bus by the increase of the reactive power consumption.

The aforementioned uni-criterion purpose functions [8], [9], [10] can provide the outcome for the multi-criterion purpose function design.

2.1.2 Matrix of sensitivity coefficients

Sensitivity coefficients31,21 are the coefficients expressing a certain relationship between the generatoric and other buses of the power system in a stable (non-dynamic) state. In general, the matrix of sensitivity coefficients is the matrix (see e.g. [15]), whose elements are represented by the coefficients of sensitivity.

Voltage sensitivity coefficients express the influence of the voltage change in a generatoric bus on the voltage change in another (or in a selected) buses in ES. For the purpose of the monograph I will further utilize voltage sensitivity coefficients.

Via these sensitivity coefficients (voltage gains) we can determine the generator bus with the largest influence on the specific bus in ES from the perspective of the best possible voltage maintenance in the bus through its load changes. Such a bus is a suitable candidate for the pilot bus function. The load bus with low sensitivity coefficients from all generatoric buses is an unsuitable candidate for the pilot bus function.

The equation is based on the calculation of the voltage coefficients matrix [4]. For simplicity I introduce the equality:

$$K = C \cdot B \, , \qquad [11]$$

where C and B are identical with [2] and [3].

The voltage sensitivity coefficient (it is a dimensionless number) is mathematically defined as the ration of the voltage change of ΔU_{Zj}, the j-th load bus [kV] and the voltage change of ΔU_{Gi}, the i-th generatoric bus [kV]:

$$k_{Uij} = \frac{\Delta UZj}{\Delta UGi} \quad i = 1, 2, \ldots n_G; \quad j = 1, 2, \ldots n_Z \, , \qquad [12]$$

where

n_G – is the number of generator buses;

n_Z – is the number of load buses, or the number of candidates for pilot buses.

Voltage changes of ΔU_{Zj} (for all $j = 1, 2, \ldots n_Z$) are acquired as the response to an artificially elicited voltage change ΔU_{Gi} in the i-th generator bus. The calculations of all voltage changes ΔU_{Zj} for the selected change ΔU_{Gi} will be executed under the condition that the voltages in other source buses remained unchanged. The change of ΔU_{Zj} in the j-th candidate bus is determined as the difference of the stable voltage ΔU_{Zj0} (before the occurrence of the voltage change in the source bus) and the stable voltage of ΔU_{Zj1} (after the voltage change in the source bus), i.e.:

$$\Delta U_{Zj} = U_{Zj1} - U_{Zj0} \quad j = 1, 2, \ldots n_Z . \qquad [13]$$

Via these obtained voltage changes it is possible to determine the sensitivity coefficients k_{Uij} among the considered generator buses and the candidates.

2.1.3 Calculation of the purpose function

Once the sensitivity coefficients matrix is developed, it is possible to calculate the purpose function. The whole calculation comprises several steps. In Step 1 it focuses on the increase (symmetric or asymmetric) of the reactive power consumption. As a result the voltage decreases U_{Gn} on the generators to the values of U'_{Gn} obtained by the calculation of the system stable state.

Since the relations among the individual generators and load buses are known from the matrix of coefficients, it is possible to calculate the voltages that have to be increased on the generators in order to balance the voltage in the pilot buses onto the original value.

If the matrix of the K system sensitivity coefficients is regular, then one solution exists for the system, i.e. the equality stands:

$$K \cdot \overline{\Delta U'_{G_i}} = \overline{\Delta U'_{Z_j}} \ , \qquad [14]$$

where

$\overline{\Delta U'_{G_i}}$ $i = 1, 2, \ldots n$ – is the vector of unknown voltages;

$\overline{\Delta U'_{Z_j}}$ $j = 1, 2, \ldots m$ – is the vector of voltage decreases on the load buses after the reactive power increase;

K – is the matrix of voltage sensitivity coefficients:

$$\mathrm{K} = \begin{bmatrix} \frac{\Delta U_{Z1}}{\Delta U_{G1}} & \frac{\Delta U_{Z2}}{\Delta U_{G1}} & \cdots & \frac{\Delta U_{Zm}}{\Delta U_{G1}} \\ \frac{\Delta U_{Z1}}{\Delta U_{G2}} & \frac{\Delta U_{Z2}}{\Delta U_{G2}} & \cdots & \frac{\Delta U_{Zm}}{\Delta U_{G2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\Delta U_{Z1}}{\Delta U_{Gn}} & \frac{\Delta U_{Z2}}{\Delta U_{Gn}} & \cdots & \frac{\Delta U_{Zm}}{\Delta U_{Gn}} \end{bmatrix} \qquad [15]$$

$\frac{\Delta U_{Zm}}{\Delta U_{Gn}}$ – is the voltage sensitivity coefficient of the load bus and the *n*-th generator;

$\Delta U_{Zm}^{'}$ – is the voltage decrease on the *m*-th load bus after the reactive power consumption increase;

$\Delta U_{Gn}^{'}$ - is the voltage to be increased on the n-th generator.

If the matrix K is regular, then the related inversion matrix exists K^{-1} of the property by which it stands:

$$K^{-1}.K = K \cdot K^{-1} = In\,, \qquad [16]$$

where *In* ones matrix of *n* degree which has the ones on the main diagonal and zeros elsewhere.

For the vector $K^{-1}.\overline{\Delta U_{Zj}^{'}}$ it then stands:

$$K\left(K^{-1}.\overline{\Delta U_{Zj}^{'}}\right) = \left(KK^{-1}\right)\Delta U_{Zj}^{'} = In\Delta U_{Zj}^{'} = \Delta U_{Zj}^{'}. \qquad [17]$$

After the multiplication [14] from the left by the matrix K^{-1}:

$$\overline{\Delta U_{G_i}^{'}} = K^{-1}.\overline{\Delta U_{Z_j}^{'}} \quad . \qquad [18]$$

Once we have calculated the voltage to be increased on the generator in order to balance the voltage on the pilot bus via the secondary voltage control, it is possible to calculate the purpose functions. Prior to the calculation it is necessary to add the calculated values of $\Delta U_{G_n}^{'}$ to the original voltage values on the generators:

$$U_{G_n} = U_{G_n}^{'} + \Delta U_{G_n}^{'} . \qquad [19]$$

The calculated voltages are added only to the generators selected for the purpose of function identification. Then it is essential to recalculate the stable ES operation so that the voltage changes on the generators occur on the load buses as well. When the required voltage deflection on the pilot bus is achieved, the algorithm is accomplished, which frequently happens after the first cycle.

Now it is possible to calculate the purpose functions, e.g. according to [8], [9], [10] for the individual pairs - generator-load bus.

2.1.4 Multi-criterion purpose function

The solution to the pilot buses selection 31 considers only one purpose function, i.e. the minimum of squared voltage deviation. To achieve the objective evaluation of the pilot buses, it is useful to deploy the multi-criterion purpose function.

Regarding the experiments 21 I can say that the purpose function selection has no crucial influence on the optimal pilot bus selection, or *n* of buses. Nevertheless, certain differences can occur in the order of the best pilot buses; therefore, it is suitable to consider the multi-criterion purpose function instead. The proposed multi-criterion purpose function regards the following functions:

- minimum of the squared voltage deviation [8];
- minimum of transferred reactive power [9];
- minimum of the maximum relative voltage deviation [10].

Via the scales the function modifies the order of individual components and subsequently it sums them. The coefficients were empirically designed so that the individual purpose functions are comparatively regarded and compensate the order of their typical output values.

$$f_K = f_{Ukv} + 0{,}01 \cdot f_{Qp} + 1000 \cdot f_{\Delta Ur} \cdot \qquad [20]$$

The function usually maintains the order of the best pilot buses obtained according to the minimum value of the squared voltage deviation; however, it considers the principal deflection changes of the transferred reactive power in the system, or the crucial voltage deviation changes by a relative low value of the reactive power transferred. The consideration of more purpose functions reflects also the concern that the use of the pilot bus with a relative low value of the sensitivity coefficient between itself and the generatoric bus can under specific circumstances in the secondary voltage control cause the consumption increase, or reactive power increase in the system.

2.2 Methods of pilot buses selection

The first solutions to the pilot buses selection 39 were based on the large enumeration of the possible solutions in terms of the selection of one or two pilot buses, so that the chosen factor achieves the minimum value. Such a solution was applied only for the smaller systems.

The technique of searching the optimization task space is not suitable for the systems of a real size as it is demanding and time consuming. In 18 and 38 they utilized the approach of the simulated annealing, which selects two or three pilot buses in order to maintain the voltage deflection on the buses at the minimum value.

The division of the power system into the control areas based on the concept of the power distance was a progress 3, 16, 23, 30, 41. Regarding the principle, the system is divided into areas, in which the pilot buses are placed in the "power center", one pilot bus per area.

The aforementioned approaches do not consider various operational conditions regarding different network topologies and load levels. The basic outputs for the solution to the issue were published in several contributions 7, 8, 10.

The utilization of "greedy" and "less-greedy" algorithms (they are graph algorithms; titles are derived from its behavior – described in Chapters 2.2.6 and 2.2.7) for the solution to the selection is shown in 10. The less-greedy algorithm of the *n*-th order is the direct superstructure of the greedy algorithm and guarantees a higher level of optimization. The result achieved via greedy and less-greedy algorithms can be improved by further investigation 10.

The contributions 7 and 8 deal with the selection procedure which comprises the combination of the greedy and the subsequent global search algorithm.

The implementation of the sensitivity threshold value coefficient 31, by means of which it is possible to significantly decrease the number of candidates for the buses, belongs to new approaches as well. The algorithm of reduction using the concept of the sensitivity coefficients matrix record via sparse matrix of sensitivity coefficients also contributed to the approaches.

In the monograph the author further deals with the detailed description of the available methods of pilot buses selection. Partial comparisons are mentioned in 9, 10, 18.

2.2.1 Exhaustive enumeration of the state space

The primary systematical approach to the pilot buses selection came out from the exhaustive enumeration of the state space (exhaustive enumeration algorithm). 39 31 utilize the title the method of global extreme – GlobEx). Only one or two pilot buses were selected so that the selected factor achieves its extreme. The method is not suitable for the power systems of a real size 9, 10. However, the search of the whole state space is the only way to guarantee the optimal pilot buses selection. It is applicable only as an offline tool and only for smaller power systems 18, 38.

2.2.2 *Simulated annealing*

The method of simulated annealing (the title originated due to the parallel with real annealing) described in 18 and 38, selects two or three pilot buses in order to maintain the voltage deflection at minimum 10. The basic notion of the method is to allow the temporary worsening of the status, which ensures the escape from the local extreme. The worsening is admitted only with a certain probability.

The simulated annealing algorithm 18 has more advantages of descent algorithms, while it is also capable to find the global maximum of the quality factor. In the beginning of the optimization process the method reflects the stochastic search of the state space (i.e. it evenly searches the whole state space); however, in the end it quickly converges to the local extreme, while the transmission between these two states is continuous. Nevertheless, the algorithm parameters setup is complicated. Through the pilot buses selection it is mainly the definition of neighboring buses and the issue of transmission. The setup of T parameter is another difficulty arising from the algorithm (this parameter represents the temperature in real annealing), which influences the initial and final state of the optimization as well as its speed. The speed is the key efficiency factor of the algorithm. It is necessary to compromise between the stagnation in the local extreme of the state space and the unproportionally long time needed for optimization. The question of the algorithm restart or the number of restarts is another difficulty, since every new algorithm start can provide a different result, i.e. a different local extreme.

The simulated annealing together with the steepest descent method and the random descent method belong to the group of algorithms which do not lead to the optimal pilot buses selection, however, they are sufficient for the needs of practice as the complete enumeration of the state space is not considered 38.

The method of simulated annealing is considered in comparison to other heuristic methods as a lengthy, clumsy and inefficient process 10.

2.2.3 Method of local search

The local search algorithm 9 starts with the group of selected pilot buses and tests all possible combinations of the given pilot bus with its neighboring load buses.

In the beginning the number of pilot buses (it is the number of buses in the primary selection) and the number of algorithm repetitions are given. During each algorithm curve, first one pilot bus is selected that will change with each of its neighboring load buses and subsequently the purpose function of the new selection is calculated. If the enumerated purpose function value is better than it is for the original bus, then the new bus automatically becomes the pilot bus. The algorithm finishes, when all the pilot buses are verified. If the selected number of algorithm repetitions is not achieved, it runs one more time.

The method is efficient at maximum in terms of the computing time; however, it requires a good initial pilot buses selection. One or two repetitions usually improve the result selection 9.

2.2.4 Method of extended local search

The extended local search is based on the local search algorithm (Chapter 2.2.3), while it reflects two levels of neighboring load buses. The result selection is usually better than by the local search; however, the algorithm is less time efficient 9.

2.2.5 Method of global search

The global search 7, 8, 9 starts with the group of selected pilot buses and tests all possible combinations of the given pilot bus with all load buses, which are not pilot ones. It is the modification or enhancement of the local search.

In the beginning the number of pilot buses (the number of buses in the primary selection) and the algorithm repetitions are given. During each algorithm curve the first one pilot bus is selected, which will be replaced with every load bus which not a pilot one and subsequently the purpose function of the new selection is calculated. If the enumerated purpose function value is better than it is for the original bus, then the new bus automatically becomes the pilot bus. The algorithm finishes, when all the pilot buses are verified. If the selected number of repetitions is not achieved, then it runs one more time – one or two search repetitions lead to the quasi optimal pilot buses selection 7.

The set of pilot buses determined by this method achieves the best possible voltage profile via control which refreshes the voltage values on the pilot buses at parallel ensuring the proportional load of control generators by the reactive power. This is the result of the purpose function definition 7, 8.

The modification of the local search is the most demanding in terms of calculations; however, it achieves the best results in the selection and only rarely remains "stuck" in the local extreme. The method again requires a good initial selection of pilot buses 9.

2.2.6 Greedy algorithm

The greedy algorithm follows the rule “take the best at any moment”. The greedy algorithm is a multi-level optimization procedure comprising the cascade sequence of search in the width. In each step the search is carried out to the width and the best solution selected. The solution becomes the outcome for the next optimization step 10.

The algorithm selects the pilot buses one by one, i.e. from the given set of pilot buses; the next pilot bus to be selected is the load bus showing the highest immediate improvement of the purpose function, if such a load bus exists. Besides, the load bus, once assigned as a pilot bus, is maintained in the set of selected pilot buses during the whole search; i.e. that for the selected load bus the purpose function is not calculated. The initial set of pilot buses is an empty set. 7, 8, 9, 10.

The algorithm utilizes two stopping criteria 7, 8. One is based on the number of selected pilot buses, determined in advanced. The other one comes from the improvement rate of the purpose function. The number of pilot buses can be determined via the use of the second criterion, i.e. the pilot buses are selected until the sufficiently low improvement rate of the purpose function is achieved.

The computational complexity for the given number of pilot buses grows linearly with the number of variables. The algorithm for the appropriate number of variables is surprisingly fast 10.

The method together with the algorithms described in Chapters 2.2.7 and 2.2.8 improves other approaches (Chapters 2.2.1, 2.2.2) for the selection in two directions 10:

- Utilizes information on the whole system due to the efficient heuristic approach, i.e. the system is not arbitrarily divided into areas in advance to lower the dimensions of the issue.
- Considers various operational conditions regarding different load levels (seasonal load changes) and various network topologies.

2.2.7 Less-greedy algorithm

The less-greedy algorithm of the *n*-th order 10 utilizes the same structure as the original greedy algorithm (Chapter 2.2.6); however, it enhances its search rules. The best *n* solutions obtained in each optimization step remember and subsequently uses them as the outcome solutions for the next step. In each step *n* searches to the width is carried out, except the first step, when only one search is executed. Therefore, in comparison to the original greedy algorithm it has approximately *n*-times higher computational complexity.

The aforementioned illustrates that the less-greedy algorithm of the *n*-th order is the direct superstructure of the greedy algorithm and is used for the local validation of the greedy algorithm since it ensures the higher optimization level in comparison to the greedy algorithm 10.

2.2.8 Two-phase technology

The method comes from the non-linear approach of the aforementioned in 7 and 8. It was utilized also in 28 and 31 with the title “Double-phase selection” technique with fixed pilot buses – AlFix.

The method comprises two phases:

1. Eliminative greedy algorithm (Chapter 2.2.6) forms the initial group of pilot buses.
2. Global search (Chapter 2.2.5) subsequently optimizes the selection.

The purpose function is the minimum value of squared voltage deviations. The procedure consists of these steps 7:

1. Consideration of the set of essential basic cases including various loads levels and various network topologies. The definition of the set of related reactive power malfunctions corresponds with individual basic cases. The set of variables to be considered is the result.
2. Generation of the pilot buses set via the greedy algorithm in the first step of the selection procedure and the global search in the second step.
3. Complete calculation of the load flow for each proposed set of the pilot buses for each case, the coefficient calculation of the average value power of the voltage value deviation for the corresponding basic case. The calculation of an average coefficient for all cases.

4. Selection of the set of pilot buses with the lowest average coefficient voltage deviation.

Eliminative algorithm

The eliminative algorithm 31 selects the pilot buses one by one, i.e. from the given subset of selected pilot buses; the following pilot bus to be selected is the load bus showing the highest immediate purpose function improvement, in the case that such a load bus exists. Furthermore, the load bus, once marked as a pilot bus, is maintained in the set of selected pilot buses during the whole search, i.e. the purpose function is not calculated for the selected load bus. The initial set of pilot buses is an empty set.

The eliminative algorithm consists of the following steps:

1. The set of selected pilot buses is an empty set and the value of the current purpose function is plus infinity.
2. The load bus from the subset of so far unselected buses is always included in the set of the pilot buses one by one and its purpose function is evaluated. The load bus with the highest immediate increase of the purpose function is added to the set of the pilot buses. This will be the current set of pilot buses. The value of the purpose function is set to the value of the related purpose function.
3. If the value of the purpose function enumerated for the set of pilot buses before Step 2 is smaller than the values of the purpose function enumerated for the current set of pilot buses, stop. The set of pilot buses before Step 2 is optimal for the eliminative algorithm. In opposite case, next.

4. If the required number of pilot buses is not achieved and the rate of the purpose function increase is not high enough, go to Step 2. In the opposite case, the current set of pilot buses is optimal.

Two finishing criteria are used. The first criterion is based on the prior determined number of pilot buses to be selected; the other criterion is based on the rate of the purpose function improvement. The second one can be used for the determination of selected pilot buses number, i.e. pilot buses are being selected until the rate of the purpose function improvement is not low enough.

The selection carried out by the eliminative algorithm is improved by the modified eliminative algorithm.

Modified eliminative algorithm

As the eliminative algorithm does not ensure the achievement of the global extreme, another method, whose result of the pilot buses selection is closer to the global extreme, has been developed. The principle 31 consists in the fact that each of the pilot buses selected by the eliminative algorithm is substituted by one of the set of the candidates for pilot buses and the purpose function is evaluated. The procedure is repeatedly activated for a specified number of times.

It is necessary to define the calculator of pilot buses in the range from 1 to number of pilot buses in the initial selection, and the calculator of restarts in the range from 1 to the required number of algorithm activations. The procedure consists of the following steps:

1. Identification of the initial selection (i.e. eliminative algorithm). The setup of the restarts and pilot buses calculators into 1.

2. We consider the pilot bus corresponding with the calculator of pilot buses. This pilot bus exchanges with each load bus which is not currently marked as a pilot bus and the value of the purpose function for each of these combinations is enumerated. Then the combination with the lowest purpose function value is selected, and becomes the current solution.
3. Increment the pilot buses counter by 1. If the current value of the calculator is smaller or equal to the required number of selected pilot buses, go to Step 2. In the opposite case, continue.
4. Increment the restarts calculator by 1 unit. If the value of the calculator is smaller than the initially determined number of restarts, set the pilot buses calculator to 1 and go to Step 2. In the opposite case, stop, the current solution is optimal.

2.2.9 Concept of electrical distance

This concept of electrical distance 3, 23, 30, 41 divides the power system into individual control areas, in which the pilot buses are placed in "the electric center", for each area it is one pilot bus. The selection of the pilot bus for each area is carried out again based on the concept of electrical distance 10.

2.2.10 Concept of decomposition – decentralized control

According to 16, various changes of the active power, reactive power, and control voltage impacts have a local influence on the power system. Therefore, it is possible to divide the power system into several control areas, in which the changes of control variables (voltages on generators) related to the given areas elicit significant voltage changes only on the pilot buses belonging to these control areas.

The advantages offered by the decentralized control system 16:

- shortening the computing time consumption;
- small requirements for computing resources;
- possibility of parallel processing of pilot buses selection;
- higher reliability.

As a consequence the control impacts of generators in the areas of decentralized control are large enough for the voltage control on the pilot buses related to this area. The malfunctions generated by the control impacts in the neighboring areas are sufficiently paralyzed by the generators controlling these areas.

It is recommended that the generators in the given control area are capable to manage the possible changes of the shunt load (transmitted onto the voltage changes on the pilot buses) by maintaining the adequate limit of the reactive power.

Due to this issue, the high number of candidates' combinations for the pilot buses is significantly reduced, since the original system is decomposed into several subsystems of a smaller size.

2.2.11 Operative selection

Besides the selection of optimal pilot buses via the coefficients of sensitivity, there are also other approaches considering various needs, e.g. the selection of pilot buses regarding their cutoff powers 6. The bus with the highest short-circuit level in the given area becomes the pilot bus. The generator is assigned to the selected bus by using the matrix of sensitivity expressing the ratios of the voltage change on the given pilot bus to the change of generator power. The generator with the highest value of the ratio is selected.

2.2.12 Methods of reduction of candidates number for pilot buses

The chapter describes the approaches to the reduction of the number of candidates for the pilot buses utilized in 21, 31.

Method of sensitivity threshold value coefficient

According to the values of the voltage coefficients of sensitivity it is possible to determine such a generator bus, which has the highest influence on the voltage of the specific bus in ES from the point of the best possible maintenance on the bus by its load changes. The bus is the suitable candidate for the pilot bus function for the generator bus given. The basic point of the method is that the load bus with low coefficients of sensitivity from all generator buses is not a suitable candidate for the pilot bus function 31.

The matrix of sensitivity contains an amount of coefficients close to zero; therefore, the threshold value σ of sensitivity coefficients for the number reduction was introduced. It is obvious that the selected highest value σ means a smaller number of candidates for pilot buses.

The method adopted is identical with the GlobEx algorithm (chapt. 2.2.1). The difference is in the added testing condition of the threshold value for each coefficient participating in the combination given.

If there is at least one coefficient in the combination which does not meet the threshold value, the combination is labeled as unsuitable and the algorithm comes to the calculation of another combination. Thus, the number of candidates, which were further elaborated in31 is significantly lowered.

Method of sparse matrix of threshold coefficients

The essential drawback of the previous method lies in the fact that even if it calculates the combinations only for the pairs meeting the condition, it still works with the whole matrix of coefficients, which hinders the calculations significantly. Regarding the fact that there are fewer members meeting the condition than the unsuitable members, the transformation of the sensitivity coefficients matrix to the reduced matrix comprising only the members meeting the condition of the threshold value was proposed. As a result, there is a multiply smaller reduced matrix containing only a fraction of members of the original matrix. 31.

In principle, it is a special type of record of a sparse matrix; which is the matrix comprising only a small percentage of non-zero elements. The principle of individual elements record of the reduced matrix of sensitivity is in the unambiguous definition of the value and position of each element in the original matrix of the sensitivity coefficients.

Method of regressive selection

The results achieved in 21 show that the candidate suitability for the pilot bus does not directly depend on the value of its coefficient of sensitivity between the load and generatoric buses. Otherwise, it does not make sense to lower the threshold value of the coefficient of sensitivity below the experimentally investigated limit of approximately 0.01, since there are no suitable candidates for the pilot buses. At the same time it was found that between the suitability of a candidate pair evaluated via the multi-criteria purpose function [20] calculated for one investigated pilot bus for the whole power system and for the size of the coefficient of sensitivity no direct monitorable connection exists. Regarding the aforementioned it is obvious that it is impossible to arrange the candidates according to their coefficients so that the primary arrangement of the suitable candidates allowed simple reduction of the number of suitable candidate pairs. Nevertheless, the reduction is necessary, if it has to meet the objective of the issue submitted – the optimal pilot buses selection for the needs of dispatch control.

Regarding this point, the following hypothesis of the regressive candidate pairs´ selection was stated:

The principle is the arrangement pilot buses candidates via the purpose function value enumerated for each candidate pair generator-load bus in the investigation of one pilot bus in the power system. Due to this, each higher *n-ple* of pilot buses comprises mainly the candidates with small purpose function value for one pilot bus.

Practically, the primary enumeration of the purpose function for each candidate pair is necessary, i.e. one pilot bus for the whole power system, which represents several hundreds of possibilities. The task is manageable without any troubles in real time and regarding the evaluation of all solutions it is possible to arrange them and based on the aforementioned hypothesis eliminate the least suitable solutions. As a result, I can only identify several tens of potentially best candidate pairs for pilot *n-ples.*

The method of regressive selection has two steps:

1. The purpose function enumeration for each pair generatoric-load bus with the threshold value of the coefficient of sensitivity $\sigma = 0.01$ by the investigation of one pilot bus.
2. The arrangement of candidate pairs according to the purpose function value and the selection of best ones which can be among the candidate for the *n-ple* of pilot buses.

Following the selection of the specific number of the best candidates regressively, a new file of candidates is formed which is used in the investigation of *n-ple* of pilot buses, where $n > 1$. In 21 the sufficient number of candidates for the regressive selection was determined to the value of 50, even if it is obvious, that this method can change for different power systems. Despite the low selected value of candidates' number, there is the supposition of finding the *n-ple* of pilot buses with the purpose function value close to the global optimum.

The number of 50 candidates was determined experimentally; when in Step 2 of the regressive selection algorithm 100 candidates were utilized. And in all of the investigated *n-ple* of pilot buses for the system (where $n = \langle 2, 10 \rangle$) there was no pilot bus candidate with the serial number close or higher than 50. All candidate pairs had a better arrangement. The results of the investigation of optimal *n-ple* of pilot buses for the power system (where $n = \langle 1, 10 \rangle$) were the same for 50 and for 100 candidates as

well. Regarding this, we can assume that no further increase of the best candidates' number would bring other results in the pilot buses selection.
The method of regressive selection was effective in the significant number reduction of the pilot buses candidates. The method itself is not suitable for the investigation of the *n-ple* of pilot buses via enumeration of the whole residual state space in real time due to quite high computational complexity, although it achieves better results than the method of the threshold value of the coefficient of sensitivity with too high value of σ, and significantly faster than by the sufficient size of σ.

2.3 Genetic algorithms

Evolution algorithms 34, 35 are applicable for the solution to the wide range of optimization tasks. It is possible, even if through the optimization issues solution no specialized methods based either on the internal knowledge of the issue or on systematic search utilizing the classical optimization approaches cannot be developed. More approaches belong to evolution algorithms, e.g. 32, evolution strategies, genetic algorithms, genetic programming, etc. The competition of potential solutions and the optimization based on stochastic changes is their basic common property.

According to 1, 15, 34, 35 there are more definitions of the genetic algorithm. The genetic algorithm and its behavior can be defined as follows:

Evolution algorithms are basically simply, however, universal and very robust numeric search methods utilizing stochastic phenomena copying the natural evolution process. Genetic algorithms are a universal stochastic search approach, which within the limited space of acceptable solutions of the given issue is capable to approximate the global extreme.

2.3.1 Principle of genetic algorithms

Darwin's theory of evolution is based on the proposition of natural selection, according to which only the best individuals of the population survive. The reproduction of two individuals gives the birth to an offspring (offspring's), which is likely to be adapted to successful survival. However, the reproduction itself is not efficient enough from the point of the origin of individuals with new properties easing thus the survival. Also the mutation is implemented in the evolution, and it is influenced randomly (positively or negatively) by the genetic material of the individuals population.

To express the evolution successfulness I will use the term "fitness". In biology fitness is a relative ability to survive and reproduce the genotype in the given environment. It will have a similar importance in the artificial life as well. The fitness will be a positive number assigned to the genetic information representing the organism. It will represent its relative successfulness to execute its tasks in the given environment and enter the reproduction.

If the term "biological individual" is substituted by the term "chromosome" representing the linearly arranged information contents of an individual – genotype, we can speak about the population of chromosomes.

They will be reproduced with the probability proportional to their fitness, while the mutations are the part of reproduction. The mutations bring new information into the chromosomes, and it can increase the fitness in the next generation. New chromosomes force the chromosomes with small fitness out of the generation. This basic reproduction cycle is constantly repeated. After a specific time the chromosomes with new properties significantly increasing their fitness are likely to occur and they force the original chromosomes without these properties out.

Crossover is one of the basic features of the genetic algorithm. Its use distinguishes the genetic algorithms from other stochastic evolution algorithms. Sexual crossover, or crossover between the chromosomes is an efficient way how to overcome the probabilistic barrier of creating the chromosomes with high information contents, and hence with high fitness. Without including the crossover into the natural selection the evolution speed is significantly lowered. The crossover is very efficient, if the chromosomes entering the process have large not overlapping parts from the target chromosome. The offspring as a result, are very similar or identical with the target chromosome.

The process of reproduction starts with the quasi random selection of two chromosomes from the population depending on their fitness – the higher the fitness value, the larger the selection probability. Reproduction is the process, in which two original chromosomes are reproduced (they create offspring's) into two new chromosomes. The reproduction includes the crossover and mutation. Both operations are carried out only with a specific probability 22. Selection is another operation. It is the selection of individuals which remain unchanged until the next generation 2.

In general, the genetic algorithm (Fig. 3) operates as follows 11, 19, 32:

1. Initialization of population – first generation of chromosomes is mostly generated at random within given boundaries.
2. The fitness of all *n* chromosomes is evaluated.
3. If the finishing criterion is not achieved, e.g. the required number of generations or a specific fitness value, the algorithm continues with the selection of parents. All individuals are mutually compared and the group going to the next generation without a change is selected (parameter *r*). The other group of individuals determined for the reproduction (*n* - *r*) is selected. The most successful individuals have the highest probability of survival; however, also the less successful have a certain chance of survival.
4. The chains in the reproduction group are paired, so that the operation of crossover and subsequent mutation can be executed.
5. Through completion of the process the unchanged part of the chromosomes joins the rest of population. Step 2 follows.

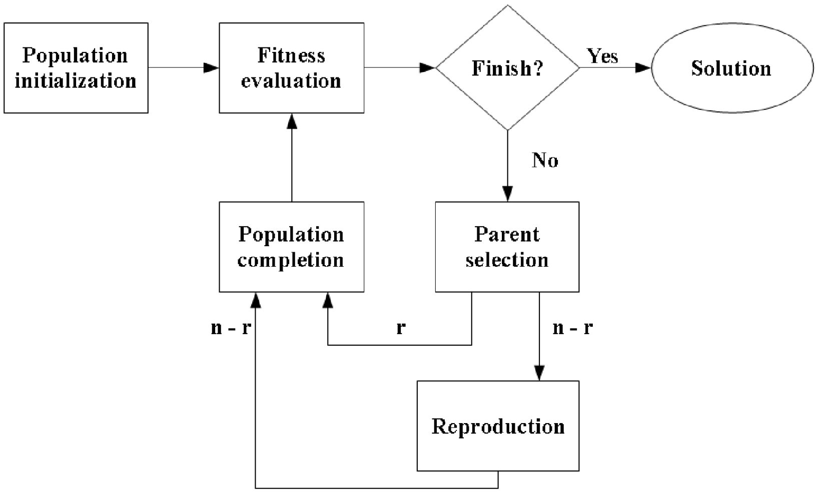


Fig. 3 *Block diagram of genetic algorithm*

The evaluation of fitness (Fig. 4) has three steps:

1. Decoding of chromosomes provides the simulation parameters.
2. The simulation or calculation is carried out.
3. Regarding the calculations the fitness of each individual is obtained.

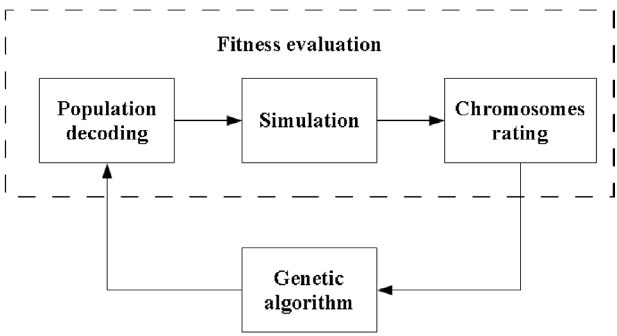


Fig. 4 *Block diagram of fitness evaluation*

If the aforementioned procedure repeats a sufficient number of times, the solution converges to global optimum 32. According to 32 the genetic algorithms differ from the majority of conventional optimizing methods by the following:

- they are able to escape from the local surroundings and approximate the global extreme;
- they carry out parallel search in several directions simultaneously;
- they do not require auxiliary information on solution development;
- they utilize stochastic phenomena considerably;
- they solve optimizing issues with tens or hundreds of variables;
- they are universal;
- they belong to more computational complex techniques.

2.3.2 Applicability of genetic algorithms

Currently the genetic algorithms have a considerably wide range of utilization, mainly through achieving the solutions to difficult issues, then for machine learning (artificial intelligence) and the development of simple programs, images or music.

Parallelism counts as their biggest advantage, since they “travel” across the searched space with a larger number of individualities (individuals), and therefore, there is a much smaller probability that they stack in the local extreme as with other methods. In addition, they are quite simple to implement.

The typical genetic algorithms are e.g.:

- search of global extremes non-linear, multi-modal issues (with many local extremes) of many variables;
- difficult combinatory or graph oriented issues;
- multi-parameter issues, by which the complexity grows exponentially or factorially;
- tasks with many limiting conditions;
- non-linear dynamic systems (prediction and data analysis);
- strategic planning, etc.

These issues can be solved by conventional methods only with difficulty, or they even cannot be solved at all.

Besides the basic versions of genetic operations (crossover, mutation) various modifications and combinations are used in the practical applications of genetic algorithms. There are several kinds of chromosomes selection into the new populations. Except the choice of specific genetic operations the overall architecture of the genetic algorithm can modify (sequence of steps). The calculation – the core of the issue investigated – of the purpose function depends on the given application 1.

The used basic operation modifications are as follows 32:

- mutations modifications – simple mutation, additive mutation with proportional and normal probability division, multiplicative mutation, mutation of duplicate chains;
- operations based on crossover – one point and multi-point crossover of two chromosomes, discrete crossover, heuristic crossover;
- methods of selection – selection of the most successful individuals, selection and classification of chains according to their successfulness,

random selection of chromosomes, selection according to maximum diversity, stochastic proportional selection, tournament selection, selection via roulette wheel weight.

Other auxiliary and special functions are e.g. the generation of random all-numeric or real numeric population, mixing of chromosomes in the population, crossover among more parents, and many others.

3. FORMULATION OF SUBJECT AND OBJECTIVES

Based on the analyses of available sources the previous chapters introduced the fundamental subject and approaches to optimal pilot buses selection of the secondary voltage control. The basic feature of the task complexity is represented by the multidimensionality of the subject.

3.1 Fundamentals

Based on the analyses carried out it stands that the pilot buses selection is the fundamental prerequisite for the correct operation of secondary voltage control of the power system. As aforementioned, there is a large amount of suitable load buses, which are part of the candidate pairs formed by generatoric and load buses. In addition, the number of potential solutions grows exponentially with the increase of the number of searched pilot buses within the system. All in all, the selection of incorrect *n-ple* of pilot buses can cause considerable economic losses.

The reactive power consumption loading the power system is not proportionally split in time or geographically, and furthermore, (e.g. the failure of the part of control with the impact on overall power system can cause significant topologic changes) we can identify another issue, which is represented by the character of the space investigated.

Based on these assumptions we can consider the pilot buses selection an essential issue of the secondary voltage control of the power system. We have to realize that the suitability of the load bus as a pilot bus of the power system varies not only via typical time changes in the system, but mainly by unpredictable changes of the load via the reactive power consumption. Therefore, it is necessary to ensure the optimal pilot buses selection for the needs of the dispatch control in the power system in real time. It practically means to develop the system of optimal selection operating with the period of several minutes.

By summarizing the aforementioned I can define the principal subject of the monograph as follows:

The design of the supportive system of the pilot buses selection usable in the dispatch control of power system capable to operate in real time and by the asymmetric load of reactive power changing in time.

3.2 Scientific objectives

Regarding the executed analyses and formulated fundamental issue, we can stipulate the following objectives of the monograph:

- Analysis of current known methods of optimal pilot buses selection regarding the suitability of methods for genetic algorithms application.
- Design of architecture of the optimal pilot buses selection for voltage dispatch control of complex power system buses in real time regarding the executed analysis.
- Application of an original approach to the solution of the issue of optimal pilot buses selection based on the parallelism principle and possibilities of algorithm grid structure utilization.
- Implementation of prototype software declaring the functionality of the architecture designed.

- As an inseparable part of the monograph, it is possible to articulate the objective to verify the designed algorithms not only from the point of their correctness, but in the context of their physical feasibility in the dispatch control under the real time conditions.

It can be seen from the list of the objectives that the centre of the overall aim is represented not in the design of a new method of optimal pilot buses selection, but in the modification of existing methods, so that the solution of the subject can be carried out via the use of parallelism principle (grid computer structure) with the aim of multiple shortening of the time necessary for calculation.

4. SYSTEM OF OPTIMAL PILOT BUSES SELECTION

The chapter deals with the design of the architecture of the system for the optimal pilot buses selection and describes the outcomes that were considered. Since the requirement is that the system operates in real time and the individual selection methods require the purpose functions calculation based on the simulation of the power system operation, I utilized the principles of parallel processing of calculations in the designed architecture. As it is an issue with room for wide-ranging analysis, I selected the genetic algorithms as the architecture basics, since here I could utilize the parallelism principles as well.

4.1 MODES Software

The software is determined for the calculation of transient responses (dynamic simulation) in power systems. The dynamic simulation comprises the following stages:

- acquisition of the network output operation;
- tuning of the output stable status;
- assignment of blocks into source buses;
- initialization of blocks models, loads, automations, logic, control transformers and central controllers P/f;
- determination of malfunctions, related impacts and scenarios design;
- simulation calculations;
- results analysis.

The library containing the ready models of generators, exciters, actuations and their generalized controllers is part of the software. It is necessary to emphasize, that the transmission phenomena are investigated from the point of the behavior of the network or power system as a whole, with the proportional accuracy of the used models. Therefore, they are usable only for aliquot small changes of operational quantities, which is important particularly for the actuations of blocks showing non-linearities of active members (thus the used constant parameters of the linearized model stand for the state of outcome and proportional deflections) and their controllers can have a variable structure depending on the operational state. The use of software is based on the following principles 24, 25, 26, 27:

- modular architecture;
- the concept of projects and cases;
- 3D data model;
- The concept of libraries models and type parameters.

Within the monograph submitted, I utilized the dynamic libraries Ust.dll and Dforrt.dll of the MODES Software. These libraries implement the functions via which it is possible to activate the power system dynamic simulation and the stable operation calculation. The simulation executed via this supportive software is necessary for the calculation of the candidate pair's purpose functions.

4.2 Grid computing

The performance of current computer technology highly exceeds the needs of common users and the utilization of the office computer is about 5%. Nevertheless, there are scientific and technical fields (weather forecast, financial market development simulation, aviation development, etc.)

requiring the execution of complex calculations needing thus larger data storing and faster networks. It is often impossible to meet this requirement within our own finances, therefore, the technology of distributed calculation systems has originated. It is called grid computing. The term was first used in 14 and in an analogy with the power system. The requirements for the calculation and data space are divided into more computers connected in the network, similarly as the requirements of the power system consumers are divided into more generators. Basically, a virtual super-efficient computer is formed (see Fig. 5), which administrates and operates the group of heterogeneous systems sharing the sources. The origin of these systems was made possible due to the fast development of the Internet and related standardization of heterogeneous systems communication. The grid can be taken as a lattice in *n*–dimensional space, where the points of intersection are sources and the joins represents communication channels.

To be able to implement the strategy, the following two conditions have to be met:

1. The applications have to allow remote controlling without a high added communication load, as well as they have to be able to operate independently at a maximum possible rate.
2. The remote computer has to be equipped with the complete hardware and software necessary for the correct operation of the application.

Within the grid it is necessary to manage the sharing of means among the virtual organization participants. In 14 the grid is defined by three characteristic properties:

- coordination of distributed sources;
- utilization of open and standard general boundaries and protocols;

- provision of non-trivial service quality.

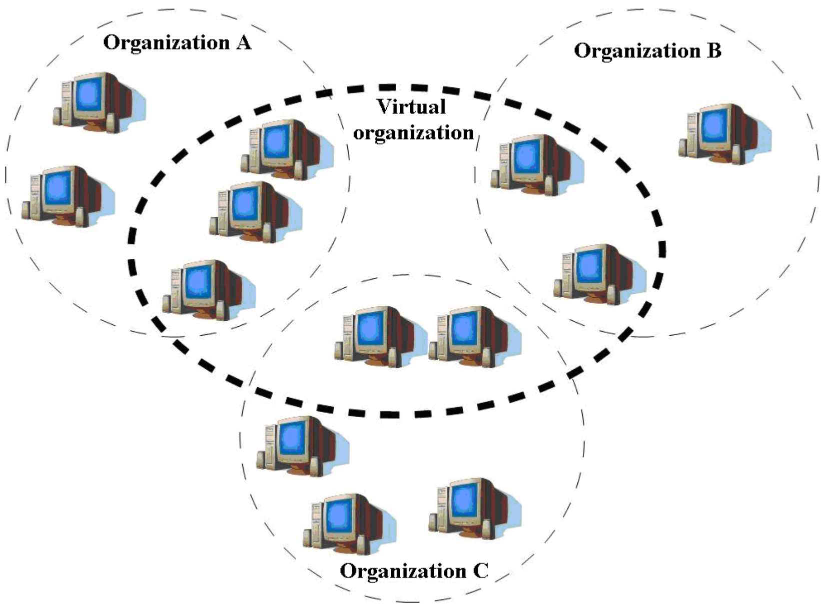


Fig. 5 *Diagram of virtual organization formed by sharing the means of real organizations*

Probably the best known grid available is BOINC (Berkeley Open Infrastructure for Network Computing), which is constantly developed at Berkeley University, USA. It is an Open – Source software, therefore, anybody can utilize or modify it 4.

BOINC utilizes the architecture client – server, where the client solves the task defined by the server and the server administrates the available sources, while the server is based on the UNIX operational system and the client is able to operate on the majority of currently available operational systems. For the project implementation in the system the standard application program boundaries are available.

The author's original intention was to utilize this system to manage the subject presented; however, the implementation of an adequate solution into the BOINC infrastructure would require a significant amount of time, or at least a group of cooperating developers. Regarding the lack of sources necessary to meet the intention, the author provides the design of their own system with the architecture reflecting the advantages of grid computing technology.

4.3 Parallel genetic algorithms

Genetic algorithms analyzed in detail in Chapter 2.3 belong to the group of evolutional algorithms and the first experiments with their implementation into the solution for the optimal pilot buses selection for the needs of the power system dispatch control were carried out in 21. Their main disadvantage was the solution slowness and the uncertainty of global extreme achievement; therefore, they were used only as a superstructure to improve the results achieved by other methods.

Anyway, the genetic algorithms have a huge potential to investigate similar issues as well as provide various possibilities of modification to fasten the operation and improve the algorithm properties in general.

In the genetic algorithm itself 33, there is the parallelism implemented within one population. By this increase of the calculation parallelism level the parallel genetic algorithm originates, which utilizes the calculation running in more independently co-existing and cooperating subpopulations. Each population lives its own life.

The probability exists that all populations will not head to the same local extreme, therefore it is suitable to utilize the migration e.g. of the best individuals among the subpopulations regarding the prior stated relations. The migration period can be variable in the range of tens or hundreds of generations and the migration itself can run periodically or can be subject to a suitable criterion, or it can be random. The migration relations do not have to connect all subpopulations; their structures can be different (see Fig. 6).

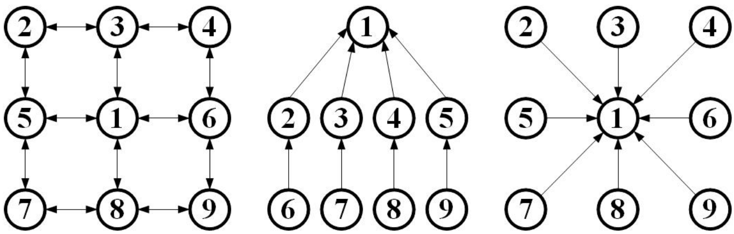


Fig. 6 *Example of various structures of parallel genetic algorithms*

Available sources 5, 33 show, that it is possible to find many recommendations for the selection of a suitable genetic algorithm parallel structure. Basically, according to 33 we can classify the parallel genetic algorithms into fine-grained (large amount of small subpopulations) and coarse-grained (small amount of large subpopulations).

4.4 Design and implementation of architecture

The architecture is based on the principle of "divide et impera", while integrating the knowledge of the aforementioned analyses. The main idea of the principle is to divide the complex task into several independent solvable operations and then decompose the operations into several computational means mutually connected within the computer communication network.

Regarding the knowledge described in Chapter 4.2, I can assume that this decomposition result will be represented by a virtual system with the computing power of much higher level when compared to the independent performance of any connected participant, i.e. the system is divided within the architecture into three parts as follows:

- GA Master – independent application – in charge of operations related to the genetic algorithm and definition of the task to be solved;
- GA Slave – independent application – in charge of operations related to fitness function value calculation of the chromosome given;
- Database – data storage and some dedicated operations processing.

The structure of the whole system of the optimal pilot buses selection for the needs of the power system dispatch control in real time is illustrated by the flow diagram in Fig. 7.

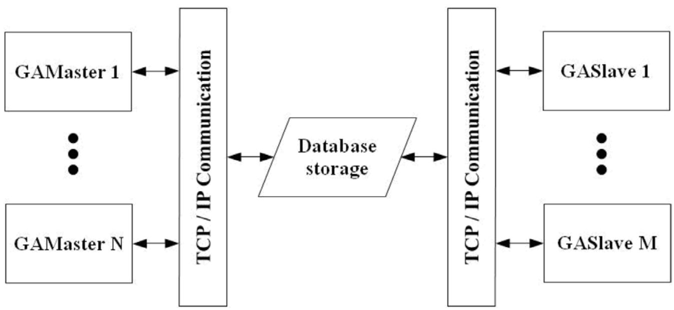


Fig. *7 Flow diagram of the designed calculation system architecture*

Blocks GA Master and GA Slave are mutually indirectly connected via the database data storage, which besides storing the individual population's chromosomes and communication mediation among other system elements, executes several tasks contributing to the increase of the calculation speed of the whole system. The whole system is described in detail in the following section.

4.4.1 Coding of the chromosome

The chromosome is the chain formed as a sequence of numeric or symbol values and represents the selected properties or parameters of an individual. It can be expressed via binary, whole numeric, real numeric, symbol values or by their combinations. The choice depends on the nature of the issue investigated 33.

Within the designed architecture the chromosome is coded via whole numeric values, while the range of values is given by the number of candidate file lines and individual genes represent the specific line of the file.

The candidate file is formed on the basis of the sensitivity coefficients matrix. Each line of the file contains the title of the generatoric bus, title of the load bus and the value of their mutual coefficient of sensitivity. The candidate file is in .csv format, which means that individual elements are mutually divided by a semicolon, and lines are finished by "&" symbol. The lines of the candidate file are arranged in a downward direction according to the value of the coefficient of sensitivity. An example of one line of the file is illustrated in Fig. 8. Indexing of the candidate file starts with the value of

0 representing the first line of the file and ends with the value (N – 1), where N is the overall number of the lines of the file.

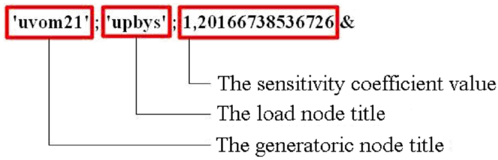


Fig. 8 *Method of coding the candidate file values*

The titles of generatoric and load buses correspond with the marking within Ust.dat file, which is one of the input files of Ust.exe program, via which the simulation of the power system stable operation runs. The simpler identification of buses within the power system model and the ease of operation with Ust.dat file is an advantage of the marking.

Due to the representation of the candidates in the file, it is very simple to apply the principles of methods to limit the number of candidates described in Chapter 2.2. For example, if we use the method of limiting the number of candidates for the pilot bus via the sensitivity threshold value coefficient, then the value of the coefficient of sensitivity in the last line is the nearest higher value, or the value equal to parameter σ. Similarly, the set of suitable candidates for the pilot bus is implemented from other methods.

4.4.2 GA Master

This independent application integrates the functions for the definition of the task investigated and the functions of genetic algorithms. The window is shown in Fig. 9.

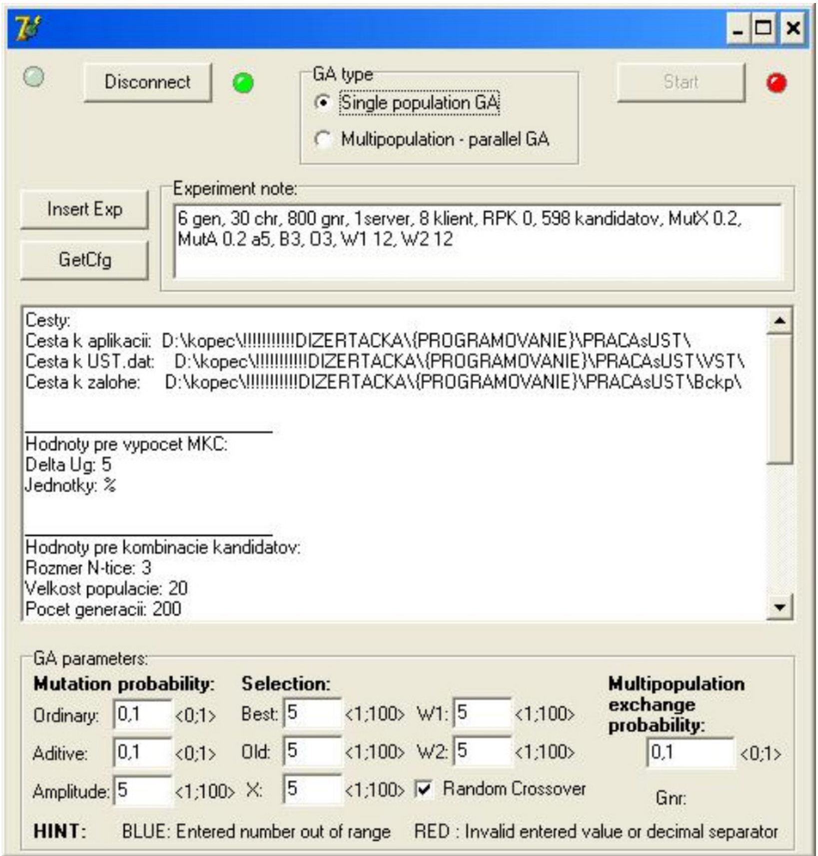


Fig. 9 *Window of GA Master application*

The definition of the task to be solved comprises two parts:

- setup of required configuration of general parameters of the task in Config.ini file;
- setup of genetic algorithm parameters for the specific subpopulation.

Subsequently, the defined task is entered into the database, while the record inserted contains the files needed for the task elaboration as follows:

- Config.ini – file containing general parameters of the task;
- Candidates.csv – candidates file;
- Ust.dat – input file of the power system model.

After the task determination it is possible to approach the system actuation, which sets off the solution of the task itself.

The following subchapters describe the functions of genetic algorithms implemented in the GA Master application and originate from 32 and 36.

fncGenSpaceI

The function provides the generation of the matrix of limitations used within other functions to define the boundaries of the set, from which the individual genes of a chromosome are selected. The matrix contains the number of lines corresponding with the population size and two columns with the saved minimum and maximum acceptable value of the gene.

fncGenPopI

The function generates at random one whole genetic algorithm population based on prior defined matrix of limitations.

fncGenerateFirstPop

This function helping to form the initialization population was developed to be able to provide the genetic algorithm the assumed suitable solutions to the task. The function first fills the population (whole or only a part) by the chromosomes read from InitPop.csv file (if it exists). If it is necessary, the residual chromosomes are complemented via fncGenPopI function, so that the defined population size is achieved.

fncSortFitI

The function is for the arrangement of the population on the basis of the fitness function value of the individual chromosomes.

fncSelBestI

The function copies the prior stated number of chromosomes with the best fitness function value from the input population into the outcome population. The chromosomes with the lowest value of the purpose function within the input population are considered best individuals.

fncSelTournI

The function implements the tournament selection of chromosomes. From the input population we form an amount of random chromosomes pairs of how many individuals we have to select into the outcome population. Subsequently, one chromosome with better (in our case lower) fitness function value is selected from each pair, while some chromosomes can be found in more pairs.

fncCrossOverI

The function forms the outcome population of the defined size by the crossover of all chromosomes of the input population, if they are of an even number. Depending on the input parameter, the parent chromosomes are selected by two ways:

- Crossover of random chromosomes comprised in the population;
- Crossover of neighboring chromosomes comprised in the population.

Similarly, the crossover is controlled by the input parameter so that it is possible to crossover one or more genes of the chromosomes selected.

fncMutXI

The function implements the normal mutation of chromosomes of the defined part of the population. Only some chromosomes are mutated depending on the mutation probability in the range of <0;1>. The chromosomes mutation is random within the values prior defined by the matrix of limitations, which is the input function parameter.

fncMutAI

This function implements the method of additive mutation of chromosomes of the defined part of the population. Only some chromosomes are mutated depending on the mutation probability in the range of <0;1>. The mutation is carried out by the addition of a random whole numeric value to the original value of the gene which is limited by the matrix of the additive mutation amplitude. This is defined in advance by the absolute values of the amplitude and enters the function as a parameter. Subsequently, the mutated value of the gene is limited by the input matrix of limitations. The probability of the mutated number random value is proportional to the space defined.

fncChckPopI

The function executes the control of the chromosomes regularity of the whole population, i.e. that the chromosomes with the length *n* has to contain exactly *n* of different genes, which follows the nature of the *n-ple* of pilot buses. It practically means that it is looked for the chromosomes comprising possible double genes (which means that one candidate for the pilot bus is on the investigated *n-ple* several times); in case of finding such an irregular chromosome the individual is replaced by a randomly generated chromosome. The function operates recursively.

fncGetBestChrmsm

This function ensures the migration of the best individuals of individual subpopulations, so that it is possible to utilize the principles of parallel genetic algorithms. The migration of the best chromosomes among the subpopulations runs randomly with the probability in the range of <0;1>, while the number of the subpopulation has to be higher than 1, i.e. if the given GA Master application instance (within own population) has the number of populations higher than 1, it is possible to achieve the reading of the best chromosomes occurring in the subpopulations of all other running application instances.

Within the GA Master application also the functions necessary for the processing of the aforementioned work files as well as the functions ensuring the communication with the database server are implemented. As a result the exchange of required data executed on the level of the database can operate; and they are called the stored procedures.

4.4.3 GA Slave

This independent application operates the fitness function calculation of the specific chromosome read from the database via the stored procedure. The application window is shown in Fig. 10. After starting the calculation the application reads the data of the running task defined by the GA Master application from the database. The parameters of the running task as well as the work files necessary for the fitness function calculation are read.

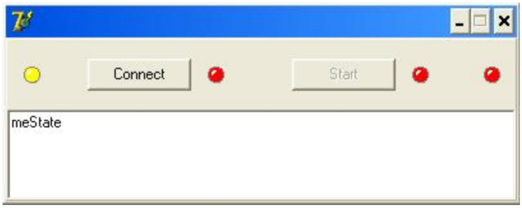


Fig. 10 *Window of GA Slave application*

Subsequently, the individual chromosomes are repeatedly selected from the database via the stored procedure, for which it is necessary to calculate the fitness function value. After the chromosomes evaluation, its corresponding calculated value is inserted into the database, and subsequently the stored function is recalled to evaluate the population. The calculation of the fitness function itself runs in several steps in terms of the procedure described in Chapter 2.1.3. The algorithm operation of the fitness function evaluation is illustrated in the flow chart in the following figure (Fig. 11).

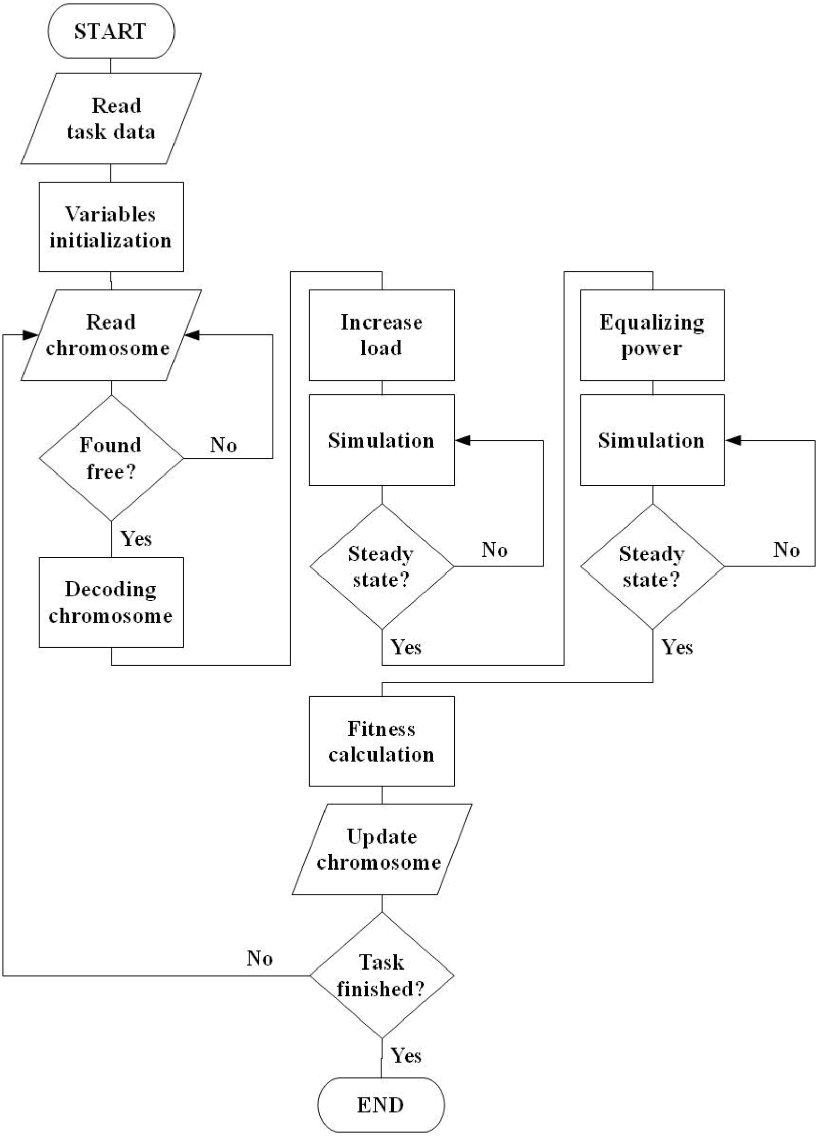


Fig. 11 *Flow chart of GA Slave application algorithm*

The reading of the chromosome to be processed runs in the cycle as seen from the flow chart in Fig. 11. Due to this the independent application operation without the necessity of intervention into its activity during the processing of the defined calculation task is ensured.

The simulation of the power system operation in both used cases is actuated in the cycle, which is given by the MODES simulation software behavior, particularly by its part UST calculating the stable operation. The number of simulations actuations was empirically stated based on the number of 10 experiments for each cycle. The development showed that if the UST program does not achieve the stable status of the power system model up to and until the 10th experiment then it does not achieve it at all. In this case, the chromosome fitness value of 999999 is used (the experiments showed that the fitness value does not approximate such high values), and subsequently the chromosome is undefined within the genetic algorithm as it is obvious that it cannot be a suitable solution to the defined calculation task.

The calculation of the fitness function value utilizes the multi-criteria purpose function defined in Chapter 2.1.4, while the weight of the individual function components are defined in the input file of the general setups of Config.ini system.

The operations of reading the chromosome from the database as well as storing the chromosome into the database utilize the stored procedures processed directly on the database server.

4.4.4 Database

The database controlling the indirect communication among the individual running instances of other system applications is the central point of the designed architecture. The database is implemented via the system of MySQL 5.1.30 database control, which is freely available for non-commercial utilization and offers sufficient means for the implementation of the architecture designed. The physical database structure is illustrated in Fig. 12.

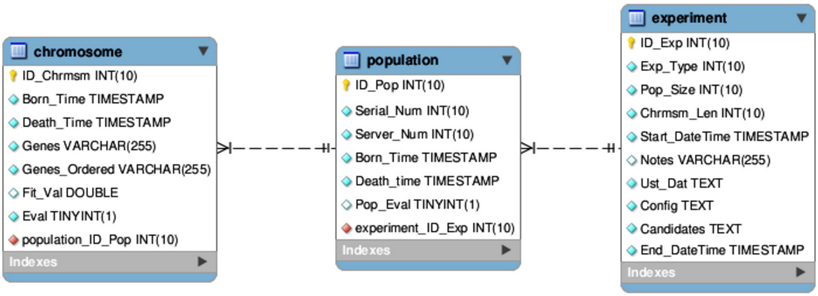


Fig. 12 *Physical model of the implemented database*

As seen from the physical model, the database consists of three tables, where there are relations 1: N having the following attributes:

1. Table ***experiment***:
   - ID_Exp – unambiguous identificator of the task solved;
   - Exp_Type – type of genetic algorithm (0-ordinary; 1-parallel);
   - Pop_Size – size of population;
   - ChrmsmLen – length of chromosome (dimension of wanted *n-ple* of pilot buses);
   - Ust_Dat – input file Ust.dat;

- Config – file of general parameters Config.ini;
- Candidates – candidate file Candidates.csv;
- Notes – note;
- Start_DateTime – date and time of task solution actuation;
- End_DateTime – date and time of task solution accomplishment.

2. Table ***population***:
   - ID_Pop - unambiguous identificator of one population of chromosomes;
   - E.ID_Exp - unambiguous identificator of the task solved (foreign key);
   - Serial_Num – order number of population (generation);
   - Server_Num – identificator of GA Master instance;
   - Born_Time – date and time of population formation;
   - Death_Time – date and time of population evaluation;
   - Pop_Eval – factor of population evaluation (0-inserted; 1-evaluated).
3. Table ***chromosome***:
   - ID_Chrmsm - unambiguous identificator of one chromosome;
   - P.ID_Pop - unambiguous identificator of one population (foreign key);
   - Genes – genes of the chromosome in the form of the chain divided by a semicolon;
   - Genes_Ordered – genes of the chromosome in the form of the chain arranged downward;
   - Fit_Val – fitness function value;

- Eval – factor of the course (phase) of chromosome evaluation (1-inserted, 2-free, 3-processed, 4-evaluated);
- Born_Time – time of fitness calculation for the start of the given chromosome;
- Death_Time – time of fitness calculation accomplishment of the given chromosome.

The bonds of the tables are ensured via the foreign keys and time attributes are provided only for the needs of experimental system verification.

Within the implementation the analysis of selection conditions of SELECT manipulation commands used for the reading of required data from the database was carried out. Regarding the analysis the indexes for individual database tables were formed to make the aforementioned commands efficient. The need of indexes development is comprehensible due to the nature of the whole system, since the number of manipulation command calls grows proportionally with the number of running instances of GA Master and GA Slave applications. In addition, the requirements for the system resources to carry out the commands increase proportionally with the increasing number of records stored in the database.

As aforementioned, on the database server level the stored procedures are processed. It is a database object not comprising the data, but the program working with the data. Due to the use of these stored procedures the communication load of the database and approaching applications is significantly limited. For instance, without the use of these stored procedures the GA Slave application would read all chromosomes forming one processed population from the database (if the size of the population is

40, it means to transfer 40 records), so that one chromosome, for which the fitness function value has to be calculated, is selected. This communication load would significantly share in the increase of the total computing time. If we use the stored procedure for the search of the free chromosome, then only one specific record is transferred at any case.

The description of the implemented stored procedures is as follows.

ChrmsmCheck

The originally stored procedure is called within the GA Master application instance immediately after inserting the generated population into the database. The identificators of the running task and population are the input parameters of the procedure.

The procedure algorithm first reads all chromosomes, which belong to one population and whose evaluation factor has the value of 1 (newly inserted chromosome). In the following cycle it selects one of the read/recorded chromosomes and based on the Genes_Ordered attribute value it searches all other populations belonging within the running task to the given GA Master application instance for the chromosome occurrence with the same Genes_Ordered value of 4 (evaluated chromosome, its record comprises the stored fitness function).

If the search is unsuccessful, then the value of the evaluation factor is set by the procedure on 2 (the free chromosome for the calculation of the fitness function value). If the procedure succeeds in finding the identical chromosome, then the processed chromosome is assigned the same fitness function value as the found individual has and the value of the evaluation factor is set on 4, and then the processing of the newly inserted chromosome is started. The algorithm continues this way until the checking of the whole newly inserted population is carried out.

The principle of the algorithm described is based on the assumption that the order of genes in the chromosome is not important, but their combination. Due to this it is possible to arrange the genes, so that the described search can be carried out. The practical contribution of this assumption is in the fact that the fitness function value of one specific chromosome as a combination of genes is calculated just once within the solution of the whole calculation task of the specific GA Master application instance. Since the fitness function evaluation is the most time consuming process in the whole calculation system, the increase of system efficiency is obvious. The more the calculation converges to the optimum, the more often the best chromosomes will repeat in the population, and the less it is necessary to compute the fitness function. Obviously, the ratio is directly influenced by the setup of parameters of the genetic algorithms functions, e.g. the high probability of mutation increases the number of newly originated chromosomes with the primary occurrence.

GetChrmsm

The purpose of this originally stored procedure is the search of the chromosome which is free for the fitness function value calculation. The identificator of the running task is the input procedure parameter; the outcome parameters contained besides the control information value of genes attribute representing the found chromosome as well.

After the call procedure by the GA Slave application occurs, it first tries to acquire the lock of the table to ensure the data consistency stored in the table. The operation is necessary since the parallelism of processes is the principle idea of the overall architecture, and therefore, the competition instances of the described stored procedure can be called at the same time. If the procedure succeeds to obtain the lock, then the table during the algorithm procedure execution is locked for all other competition instances due to which it cannot reach the situation when e.g. two different GA Slave applications instances evaluated the same chromosome. If the procedure does not acquire the lock, the GA Slave instance waits for a random long time and repeatedly calls for the procedure.

The procedure algorithm searches the chromosome with the value of the evaluation factor 2 (the chromosome free for the calculation of the fitness function value) and the lowest corresponding value of the chromosome identificator (by this the possibility of random not finding any of the chromosomes is eliminated). If the suitable free chromosome is found, the value of its evaluation factor is changed to 3 (chromosome being processed) and its parameters are given to the GA Slave instance for the calculation of the fitness function value. Finally the stored procedure frees the lock of the database table and allows the other competition instances the access to chromosome data.

PutChrmsm

The originally stored procedure which on the basis of input parameters inserts the calculated fitness function value of the specific chromosome into the database record, and subsequently sets up the value of the evaluation factor on 4 (evaluated chromosome whose record comprises the stored fitness function value). Finally the stored procedure is called to check the population evaluation.

PopCheck

This originally stored procedure is in charge of the specific population evaluation. The procedure is called after each insertion of the calculated fitness function value of each chromosome belonging to the population being checked.

The algorithm of the procedure selects all chromosomes belonging to the population being checked and compares the number of chromosomes having the value of the evaluation factor of the chromosome 4, with the total number of chromosomes in the population, i.e. with the defined population size. If these numbers are identical, then the population being checked is considered as completely evaluated and the evaluation factor of the population Pop_Eval is the stored procedure set on the value of 1.

The related GA Master application instance, to which the population belongs, cyclically checks the evaluation factor of the population. At the moment, when the population is evaluated, it comes to the reading of all chromosomes parameters belonging to the given population by the GA Master application instance. Subsequently, the operations of the genetic algorithm are carried out, by which the new population inserted then into the database for further processing is generated.

Except the aforementioned procedures within the GA Master and GA Slave applications, there are also some other commands for data manipulation, whose use has an auxiliary character and their communication load is not very significant, are implemented. They are commands controlling the system operation mediating the indirect communication among the subject applications instances. For example, the GA Slave application instance checks the state of the running calculation task via the manipulation command SELECT. If the instance finds that the calculation task is accomplished, the algorithm jumps out of the cycle of searching for the free chromosomes to be evaluated and waits for the new task. The checking runs on the basis of End_DateTime attribute from the Table *experiment*, i.e. only one value is transferred, from which the communication load of the command is obvious.

All other implemented auxiliary commands work at the maximum with one record from the Table *experiment*, which mediates the subject communication and regarding its structure it is obvious, that the volume of records does not grow as fast as the other two tables, and therefore, it has only a low significant system load.

This means that the database plays a key role in the architecture designed.

4.4.5 Conclusion

The previous chapters described three main parts of the designed architecture, in particular:

- GA Master application;
- GA Slave application;
- database.

Only the most important original functions, procedures and essential properties of these parts were described, so that the reader can understand the functioning of the whole architecture at his/her best.

The description shows that the principles of parallel processing of information and implementation of parallel genetic algorithms are utilized. It is the parallel processing of chromosomes, i.e. parallel calculation of their fitness function value. Depending on how many instances of GA Slave application are set up at the same time, i.e. how many instances participate in the task calculation that many chromosomes will be parallely processed. Since the fitness function evaluation was identified as the most time consuming task of the total calculation, it is obvious that the more instances of GA Slave applications are used, the more significant the overall time savings within the whole defined calculation task are.

Similarly as by the GA Slave application, the GA Master application allows parallel actuation of several instances, due to which the implementation of parallel algorithms is ensured. If we want to solve the calculation via the parallel genetic algorithm, we simply define it by means of one GA Master instance as "Multipopulation" and insert with other parameters into the database. Subsequently, we actuate that many further instances of GA Master application, how many subpopulation we want to use, while the genetic algorithm's behavior for the specific subpopulation is set up via the parameters of the specific related instance. The relationship of the subpopulations is equivalent, i.e. the migration occurs among all subpopulations.

Implementation of the architecture as described is illustrated by the flow diagram of the system (the diagram shows just one instance of both applications) in Fig. 13.

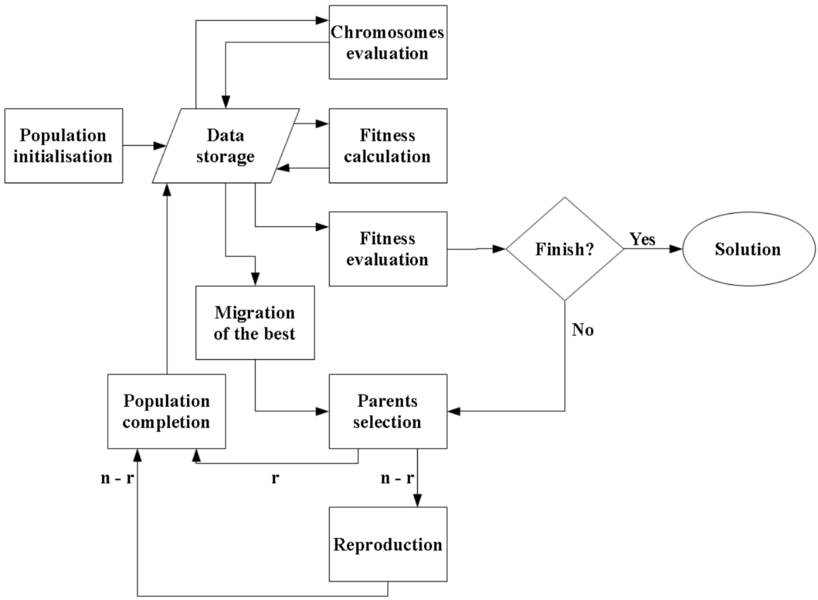


Fig. 13 *Flow diagram of parallel genetic algorithm implemented in the architecture of the system*

Due to the parallel approaches used the designed architecture considerably increases the total speed of processing of the calculation and increases the system ability to converge to the global optimum. By the use of the designed architecture on its own or in the combination with some of the methods for reducing the number candidates for pilot buses we approximate the requirement for the operation in real time. As it will be shown in the experimental part of the monograph, the implemented system is capable of improving the performance of some methods described in Chapter 2.2, which is due to the suitable implementation of the designed architecture.

5. EXPERIMENTAL VERIFICATION OF SYSTEM DESIGNED

For the experiments purposes I utilized the model of stable operation of the Slovak power system. The model on the level of 400/200 kV contains 82 branches and has 59 buses, 16 generatoric and 43 load buses.

The matrix of voltage coefficients of sensitivity was for all tests calculated for the voltage change of +5 % of the nominal voltage value on all generatoric buses. Furthermore, I used the symmetric load of the system by the reactive power consumption of +15 % of the immediate value of the reactive power consumption of individual load buses. These values are the same as in 21 and 31, so that it is possible to compare the results.

5.1 Definition of state space size

As aforementioned, the size of the state space to be searched is one of the biggest issues in searching the optimal pilot bus, or the optimal *n-ple* of pilot buses. The size of this state space is given by the number of generatoric and load buses, which are in the investigated power system and from which the candidate pairs for the pilot bus selection are formed.

The power system model used for the requirements of the monograph comprises 16 generatoric buses and 43 load buses. The size of the state space for searching just one optimal bus is defined by the multiplication of these two numbers:

$$V_P = N_G * N_Q = 16 * 43 = 688 , \qquad [21]$$

where:

V_P - size of the searched space;

N_G - number of generator buses;

N_Q - number of load buses.

It is obvious that the basic searched space is made by the number of 688 possible candidate pairs. The size of this space is the outcome parameter for the determination of the size of state spaces to be searched, if the optimal *n-tuple* of pilot buses has to be found (for n > 1), since it forms the basic set of candidates for a pilot bus. Through the number increase of the searched pilot buses within one optimal *n-tuple* it will be the combination of the *n-th* class (*n* represents the size of the searched *n-tuple*) of *k*-elements (*k* represents the size of the basic set of candidates for a pilot bus) without repetition. This dependence can be mathematically expressed as follows:

$$C_n(k) = \frac{k!}{(k-n)!.n!} \quad , \qquad [22]$$

where:

$n \leq k$;

n - dimension of searched *n-ple* (number for the set of searched pilot buses);

k - size of the set of candidates (number of candidates for one pilot bus).

From the relationship [22] it is clear that the higher the *n* and *k* parameters are, the larger the searched state space is. Since the increase of the *n* parameter, i.e. the size of the optimal combination of pilot buses is significant from the point of the investigated power system control, and then the increase is desirable. Therefore, the only way how to lower the size of the searched state space is to reduce the number of candidates for the pilot bus, i.e. to lower the k parameter. The reduction of the number of candidates for the pilot bus can be carried out by means of one of the methods described in Chapter 2.2.12. The influence of the dimension of the searched optimal *n-ple* of pilot buses and reduction of the number of candidates for the pilot bus is shown in the following tables.

SIZE OF THE SEARCHED SPACE
(number of candidate *n-tuples*) Table 1

k \ n	**Used method**	
	Global search	Regressive selection
	688	**50**
1	688	50
2	236328	1225
3	54040336	19600
4	9254407540	230300
5	1266002951472	2118760
6	144113335975896	15890700
7	14040756447937300	99884400
8	1195219392630660000	536878650
9	90305465220983400000	2505433700
10	6131741088504770000000	10272278170

SIZE OF SEARCHED SPACE (number of candidate *n-tuples*) Table 2

k / n	Used method		
	Threshold value of coefficient of sensitivity		
	$\sigma = 0.001$	$\sigma = 0.\ 01$	$\sigma = 0.3$
	363	**228**	**31**
1	363	228	31
2	65703	25878	465
3	7906261	1949476	4495
4	711563490	109658025	31465
5	51090258582	4912679520	169911
6	3048385428726	182587922160	736281
7	155467656865026	5790645531360	2629575
8	6918310730493660	159966582803820	7888725
9	272888923258361000	3910294246315600	20160075
10	9660267883345980000	85635443994311600	44352165

Tables 1 and 2 show a considerable growth of the searched space size, which is not insignificant. Similarly we can see the efficiency of the selected methods for the pilot buses candidates' reduction to limit the size of the searched space. For comparison the following methods were selected:

Global search of the operational space

This method ensures the finding of the global extreme; however, the size of the operational space is at its maximum, since all existing combinations within the power system being processed are verified. It was selected as a reference for other methods of monitoring.

Method of regressive selection

This method has been developed within 21, where the results of the method were also experimentally verified. The method considerably reduces the searched state space and provides good results. The utilization of the method of the global search for n = 1 for the initialization is its main disadvantage.

Reduction by the threshold value of the coefficient of sensitivity

This method has been developed within 31 and is able to reduce the size of the searched state space significantly. It comes from the assumption of existing dependence between the value of the coefficient of sensitivity and the purpose function value of the searched *n-tuple* of pilot buses. It was shown that the sensitivity threshold value coefficient of $\sigma = 0.3$ stated in 31 is not correct, which is proved by the executed experiments described in Chapter 5.3.

Better illustration of the state space dimensions and the efficiency of methods selected for its reduction in 0 and 0 are shown via the graph in Fig. 14. To be able to show all the data in the common graph, for the vertical axis representing the size of the state space the logarithmic scale was chosen.

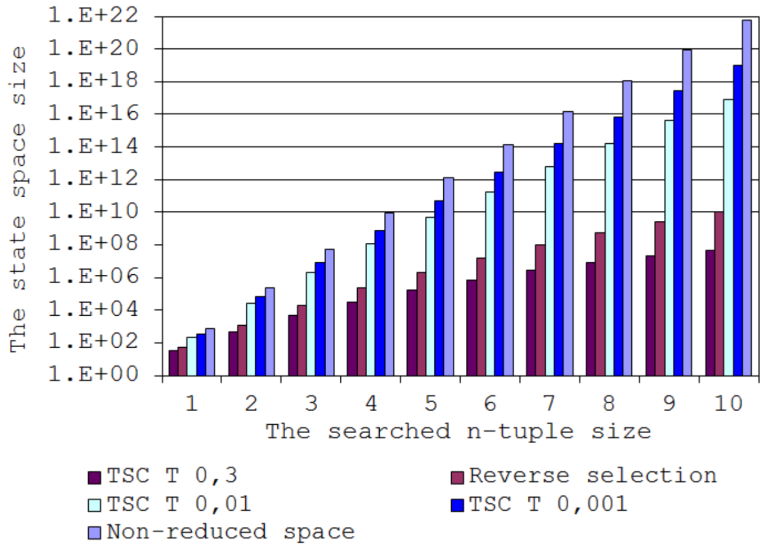


Fig. 14 *Graph of the size of searched state space for selected methods*

The enumeration of the size of the state space is a suitable metric to consider the complexity of the issue investigated.

5.2 Verification of the contribution of parallel chromosomes processing

The experiment purpose is to show the reason for the utilization of parallel processing of the fitness function value calculation in the architecture designed. As described in Chapter 4.4, a considerable speed increase of the defined task processing is expected from the use of the parallel chromosomes processing, i.e. shortening of the total computing time. To prove these affirmations several tests were executed, whose results are described in the following part.

Test No. 1

In this test a global search of the whole state space in two configurations was carried out. For the task definition I utilized the possibility to use the initialization file for the first population of the genetic algorithm InitPop.csv filled by the genes representing all candidate file Candidates.csv, which covered all the state space of the model power system.

The parameters of both configurations as well as the total time of the defined task solution are shown inTable 3. The size of the state space was 688 combinations.

EVALUATION AND PARAMETERS OF TEST No. 1 Table 3

		Conf. 1	**Conf. 2**
Number of modules	**GA Master**	1	1
	GA Slave	1	8
Average time of one chromosome fitness evaluation [s]		2.04	1.77
Time consumption of the whole task [h:m:s]		**1:50:06**	**0:16:44**

The test evaluation unambiguously proves the hypothesis correctness, that the use of the parallel approach for the fitness function calculation of individual chromosomes ensures the considerable shortening of computing time. As seen in 0, the defined task in the case of using the second configuration was solved in approximately 17 minutes. As the computing time is regular in minutes, it has the capability to implement the designed architecture for the needs of the secondary voltage control in real time (the control on this hierarchic level runs with the period of regular minutes). Obviously, there is the possibility to implement more instances of GA Slave applications for the defined task solution, which could achieve further shortening of the computing time. Nevertheless, the number of instances participating in the defined task solution will strongly depend on the configuration, particularly on the size of the genetic algorithm population, the number of running instances of the GA Master applications and the limitations of the database control system.

The executed test also showed the improvement of the method of regressive selection developed within 21, as this method utilizes the global search of the state space for one pilot bus in the first phase ($n = 1$).

Test No. 2

The calculation task defined in this test worked again with the complete unreduced candidate file, i.e. the size of the searched state space was 644 possible combinations of the generator – load buses. The working space was searched via the genetic algorithm and the optimal triplet of pilot buses was searched ($n = 3$).

The task was setup in three configurations, while it was gradually solved via using 1, 4 and 8 instances of the GA Slave application. The parameters and the evaluation of the task computing times are shown in 0. The genetic algorithm was in all three cases set up for the period of processing of 300 generations.

EVALUATION AND PARAMETERS OF TEST No. 2 Table 4

		Conf. 1	Conf. 2	Conf. 3
Number of modules	**GA Master**	1	1	1
	GA Slave	1	4	8
Time consumption of the task after 100 generations [h:m:s]		1:20:43	0:15:56	0:12:32
Time consumption of the task after 200 generations [h:m:s]		2:21:41	0:29:34	0:23:32
Time consumption of the task [h:m:s]		**2:58:14**	**0:44:01**	**0:34:29**

As seen from the results shown inTable 4, the increase of instances of the GA Slave applications, which participated in the defined task solution, and meant again the shortening of the computing time even after the first 100 generations. The test proves the results achieved in Test No. 1 and unambiguously proves the hypothesis correctness, that the use of a parallel approach for the fitness function calculation of individual chromosomes ensures the considerable shortening of the computing time and hence the total time for the solution of the defined task as well.

5.3 Sensitivity threshold value coefficient

The results of Test No. 1 (see Chapter 5.2) achieved by the global search also proved the conclusions of the tests executed within 21, where the author assumed that the threshold value of the coefficient of sensitivity $\sigma = 0.3$ stated in 31 is not correct. Since the set of candidates for the pilot bus, which was reduced via this method, provides suboptimal solutions, whose coefficient of sensitivity is deeply below this limit, however, the fitness function value significantly approximates the global optimal solution.

These affirmations prove the results included in 0. Regarding the largeness of the searched state space, the table shows only selected data whose value should be sufficient for the confirmation of the assumption articulated. Particularly, they are the parameters of the best 60 candidates evaluated and arranged upward according to the fitness function value.

The globally optimal solution is represented by the candidate with the lowest fitness function value, i.e. the candidate no. 34 is the optimal pilot bus. The number (No.) represents the order of the candidate pair in the candidate file. The values of the coefficient of sensitivity below the stated threshold limit of the coefficient of sensitivity $\sigma = 0.3$ are highlighted.

VALUES OF FITNESS AND RELATED COEFFICIENTS
OF SENSITIVITY FOR n = 1 Tabe 5

No.	Value of fitness	Coefficient of sensitivity
34	50.1937805454546	0.405040504050405
122	52.8442192727273	0.067376830892144
22	52.8854880000001	0.569462938304483
40	53.0696234545455	0.326853084775854
21	53.8283045454545	0.609725685785536
13	53.8861633636363	0.695960940967599
14	53.8862743636363	0.695872170439414
10	53.8882343636363	0.696227252552153
11	53.8882343636363	0.696138482023968
12	53.8882343636363	0.696138482023968
36	53.9146508181818	0.356152420963193
47	53.9266726363637	0.281246882793017
15	54.2407993636364	0.695694629383045
16	54.2609923636363	0.681479866320587
8	54.3346943636364	0.855860349127182
7	54.3384733636364	0.856009975062344
17	54.3612457272728	0.679970312100335
55	54.3782730000000	0.202628434886499
37	54.4641802727273	0.352169576059850
218	54.6017639090910	0.025232974910394
63	54.6828489090909	0.175866188769415
180	54.7234100000001	0.035746714456392
151	54.9055806363637	0.042741165234002
29	54.9073336363637	0.496158793471719
239	54.9626324545454	0.019440745672437
396	54.9773653636364	0.003060306030603
9	55.0100937272728	0.758930276981853
6	55.0516870000000	0.859691128433712
127	55.0687506363636	0.066491144088080
74	55.1475005454545	0.148280802292264
23	55.2043089090910	0.563753581661891
110	55.3271471818182	0.087550776583035
60	55.4260078181818	0.181953087601723

No.	Value of fitness	Coefficient of sensitivity
52	55.4273228181818	0.210196266156056
70	55.5518556363637	0.158321434050437
68	55.6648981818182	0.158415841584158
69	55.6691911818182	0.158368637817298
115	55.7725586363636	0.077745940783190
73	55.8387664545455	0.148444231689804
64	55.9156160909092	0.172456140350877
81	55.9156160909092	0.135964912280702
26	55.9398022727272	0.550872162485066
117	55.9403520909091	0.075563546109846
5	55.9528521818182	0.903247373447947
61	56.0289969090909	0.180000000000000
78	56.0318059090909	0.141578947368421
62	56.0399489090909	0.179824561403509
77	56.0408179090909	0.141578947368421
253	56.0587663636364	0.017017543859649
233	56.0597663636364	0.021754385964912
27	56.0631927272727	0.515222594542843
20	56.0937311818182	0.655869496038990
19	56.1446064545455	0.668933290903914
18	56.1456184545455	0.669100775453464
4	56.2221306363637	0.960021438022342
2	56.2266120909091	1.000000000000000
3	56.2309440909091	0.999988414259729
0	56.2327600909091	1.201667385367260
1	56.2327600909091	1.194627036296080
24	56.2327600909091	0.560586459680897

The graph illustrating the data included in Table 5 is in the part “A” and complements the overall illustration of the subject described.

5.4 Task processing via genetic algorithm

The purpose of this experiment is to consider the main properties of the implemented genetic algorithms, particularly the computational complexity and the speed of the algorithm convergence to the optimal solution.

For the classical genetic algorithms (see the flow diagram in Fig. 3) the computational complexity can be determined quite simply as the product of the population size and the number of generations. The size of population in this case is represented by the number of setups of the fitness function value calculation within one generation.

However, this calculation cannot by applied for the architecture designed (see the flow diagram in Fig. 13) regarding the database utilization due to which it is possible to implement the functions ensuring that the fitness function is evaluated just once for just one chromosome (for details see Chapter 4.4.4). Therefore it is not possible to define how many setups of the fitness function value calculation run within one population.

Regarding the aforementioned, the average time period of one population processing will be used for the determination of computational complexity of individual experiments, by which it is possible to obtain an approximate idea of the speed of the defined calculation task processing, i.e. of the speed of the system implemented.

The determination of the number of generations necessary for the achievement of the fitness function global minimum is another issue. The number can be determined experimentally on the basis of the speed of genetic algorithm convergence to the global optimum. However, the convergence speed strongly depends on many factors. The algorithm structure itself has a significant influence, i.e. not only the functions used within, but the arrangement of these functions as well. Other factors considerably influencing the convergence speed are as follows:

- size of population;
- number of the best chromosomes entering the new population;
- number of original individuals entering the new population;
- the method of original individuals selection;
- size and selection method of the chromosomes work group, on which the genetic operations are executed;
- probability value of crossover, mutation and migration.

During the experiments the convergence speed of the genetic algorithm for various setups of the mentioned parameters was monitored. The structure of the used algorithm was implemented in terms of the flow diagram in Fig. 13 and for the fitness function value calculation of the multi-criteria function according to the relationships [20] was used. The individual populations were generated at random without the initialization. Regarding the large amount of data, in the following experiments only the graphs are shown, the tables of values regarding which these graphs were made are not included.

5.4.1 Searching for one optimal pilot bus (n = 1)

The graph in Fig. 15 shows the convergence of the genetic algorithm outcome to the global optimum of one pilot bus. The value of the global optimal solution of the multi-criterial function for this case is

f_K = 50.1937805454546. The setup of Tests No. 1 to 4 were as follows: number of generations = 200, size of population = 30, number of the best individuals = 5, number of original individuals = 5, size of the work group = 20 (10 of the best ones, 10 of the original ones), probability of normal mutation = 0.1, probability of additive mutation = 0.1 and amplitude of additive mutation = 5. The searched space was reduced via the method sensitivity threshold value coefficient of σ = 0, i.e. its size was 598 candidates. As shown in the following graph (Fig. 15), the algorithm needed for the convergence to the optimal solution is less than 50 generations.

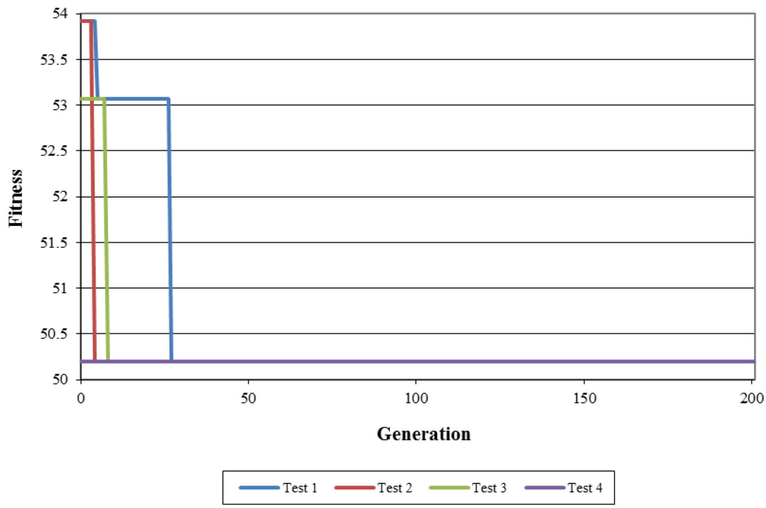


Fig. 15 *Convergence of calculation for n = 1*

Another experiment comprised 4 tests, within which the influence of the genetic algorithm parameters change on the convergence speed was compared. Test No. 1 has the same setup of parameters and as in the previous experiment as seen in the graph in Fig. 16 this prematurely converged to the value of the multi-criterial purpose function f_K = 52.8442192727273 (the value with the second best fitness function value in the order). Therefore, the parameters of Tests 2 to 3 were modified as follows: number of generations = 300, size of population = 30, number of the best individuals = 3, number of original individuals = 3, size of work group = 24 (12 of the best ones, 12 of the original ones), probability of normal mutation = 0.2, probability of additive mutation = 0.2 and amplitude of additive mutation = 5. The searched space was reduced via the method of the sensitivity threshold value coefficient of σ = 0, i.e. its size was 598 candidates. As shown in the following graph (Fig. 16), the algorithm needed for the convergence to the optimal solution for the determined parameters setup of less than 10 generations. Regarding the transparency, the graph shows only the first 60 generations as the fitness function value did not change until the end of the experiment.

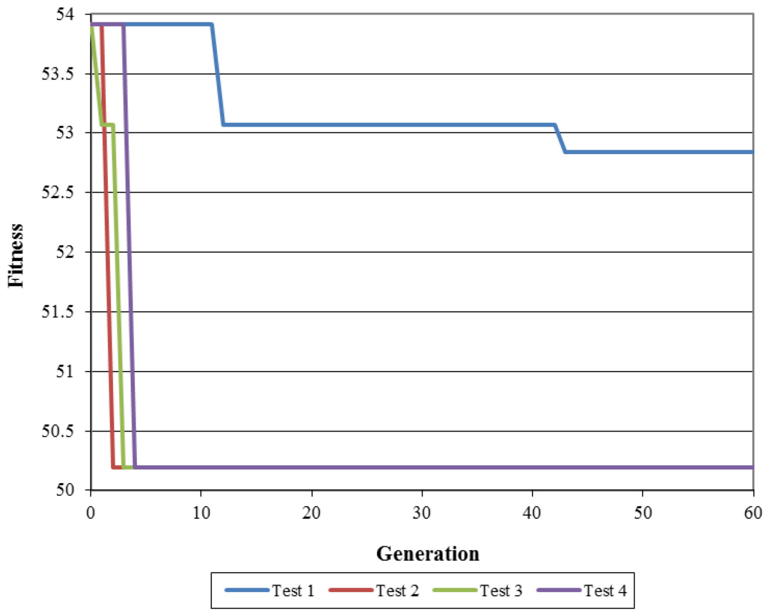


Fig. 16 *Convergence of calculation for n = 1*

The average time of processing of one population obtained from the data for the tests executed within the first experiment was 46.74 seconds, which shows that the average time of the defined calculation task processing is equal to 46.74 * 200 = 9 348 seconds, which makes 2h 35m 49s. In the tests only one instance of the GA Slave application was used.

The average time for one population processing for the tests executed within the second experiment and obtained in the same way as in the previous one, is 4.33 seconds, which shows that the average time of the defined calculation task processing is equal to 4.33 * 300 = 1 299 seconds, which makes 21m 39s. For the tests, four instances of GA Slave application were used.

The comparison showed an obvious advantage of the parallel approach for the calculation of the fitness function value as well as the considerable influence of the genetic algorithm parameters on the convergence speed to the global optimum.

5.4.2 Searching the optimal pair of pilot buses (n = 2)

The following graph (Fig. 17) evaluates two tests within this experiment. For better transparency only first 100 generations are illustrated.

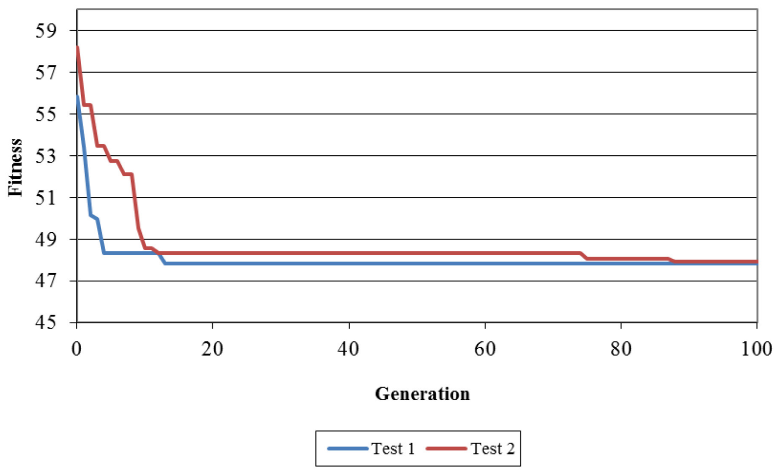


Fig. 17 *Convergence of calculation for n = 2*

The parameters of the genetic algorithm setup for both tests were the same as the previous experiments, in particular: number of generations = 300, size of population = 30, number of the best individuals = 3, number of original individuals = 3, size of work group = 24 (12 of the best ones, 12 of the original ones), probability of normal mutation = 0.2, probability of additive mutation = 0.2, amplitude of additive mutation = 5, and number of random crossover = 1. The searched space was reduced via the method of the sensitivity threshold value coefficient of σ = 0, i.e. its size was 598 candidates, which corresponds to the value of 178 503 combinations. In Test No.1 the global optimum was achieved in the 13th generation and in Test No.2 it was achieved in the 122nd generation. The result converged to the global optimal value of the fitness function f_K = 47.8196342727272.

The average time of one population processing obtained from the data for these tests executed within the experiment described was 13.8 seconds, which shows that the average time for the defined calculation task processing is equal to 24.2 * 300 = 7 260 seconds, which makes 2h 10m 12s. In both tests, two instances of GA Slave application were used.

5.4.3 Searching the optimal triplet of pilot buses (n = 3)

The following experiment comprises 5 tests with the same genetic algorithm parameters, while the number of instances of the GA Slave application participating in the defined task calculation was changed. The tests parameters were as follows: number of generations = 300, size of population = 30, number of the best individuals = 3, number of original individuals = 3, size of work group = 24 (12 of the best ones, 12 of the original ones), probability of normal mutation = 0.2, probability of additive mutation = 0.2, amplitude of additive mutation = 5 and number of points of random crossover = 1. The searched space was reduced by the method of the sensitivity threshold value coefficient of $\sigma = 0$, i.e. its size was 598 candidates, which corresponds to the value of 35 462 596 combinations.

AVERAGE COMPUTING TIMES FOR TEST No. 1 TO 5 Table 6

Number of GA Slave modules	**Average time of one population evaluation**	**Number of generations**	**Average time consumption of the task**	
[pcs]	[s]	[pcs]	[s]	[h:m:s]
1	44.04	300	13212	**3:40:12**
1	44.52	300	13356	**3:42:36**
4	8.79	300	2637	**0:43:57**
4	8.22	300	2466	**0:41:06**
8	6.42	300	1926	**0:32:06**

The average times of one population processing obtained from the data for these tests executed within the experiment described and the values of the total average time are shown in 0, where the comparisons clearly prove the advantage of using the parallel approach for the fitness function calculation.

The algorithm convergence within the individual tests can be considered on the basis of the graph in Fig. 18.

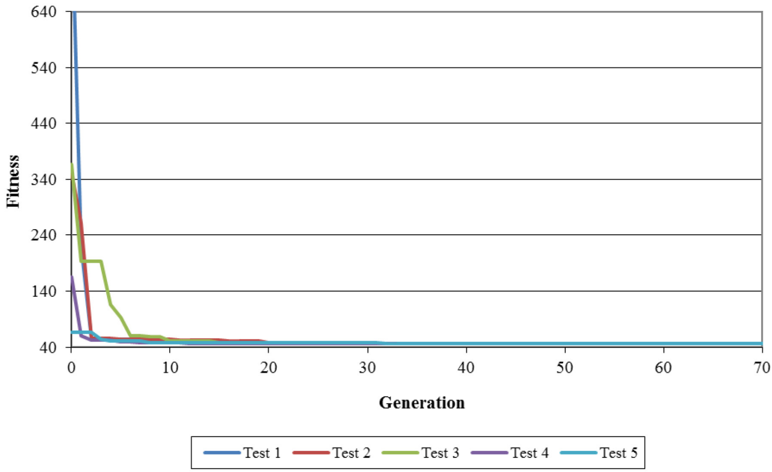


Fig. 18 *Convergence of calculation for = 3*

The result achieved in tests no. 1, 2, 4 and 5 converged to the global optimal fitness function value f_K = 45.7586855454545, convergence ran quite fast. Except the tests no. 3 and 4 the optimum was achieved sooner than in the 140th generation. In case of test no. 3 the result prematurely converged to the value f_K =45.7959084545455, while in the number of determined generations it could not leave the local optimum. In the case of Test No. 4 it also reached temporary stagnation in the same local extreme as in Test No. 3, finally the algorithm peaked on the extreme left and converged to the global optimum in the 219th generation. For better transparency the graph shows only the first 70 generations.

The difficulty with the convergence of the defined calculation task could be eliminated by further tuning of the genetic algorithm parameters.

However, this requires the execution of a large amount of experiments with careful monitoring of the individual parameters influence.

5.4.4 Searching of the optimal sextuplet of pilot buses (n = 6)

Within this experiment the tests of the algorithm convergence to the global optimum with the following parameters setup for the individual groups of tests.

Tests No. 1, 2 and 3: number of generations = 800, size of population = 30, number of the best individuals = 3, number of original individuals = 3, size of work group = 24 (12 of the best ones, 12 of the original ones), probability of normal mutation = 0.2, probability of additive mutation = 0.2, amplitude of additive mutation = 5, and number of points of random crossover = 3. The searched space was reduced via the method of sensitivity threshold value coefficient of $\sigma = 0$, i.e. its size was 598 candidates, which corresponds with the value of 61 936 603 045 317 combinations. Number of the instances of GA Slave application participating in the calculation = 8.

Tests No. 4 and 5: number of generations = 800, size of population = 40, number of the best individuals = 2, number of original individuals = 4, size of work group = 34 (20 of the best ones, 14 of the original ones), probability of normal mutation = 0.25, probability of additive mutation = 0.25, amplitude of additive mutation = 3, and number of points of random crossover = 3. The searched space was reduced via the method of sensitivity threshold value coefficient of $\sigma = 0.01$, i.e. its size was 312 candidates, which corresponds with the value of 220 651 676 244 combinations. The value was used on the basis of recommendations mentioned in 21. The number of instances of the GA Slave application participating in the calculation = 8.

In all five tests the other fitness function value was achieved as shown in the following table (Table 7), therefore, it is not possible to determine the global optimum of the task.

FOUND CHROMOSOMES WITH RELATED FITNESS FUNCTION VALUE

Table 7

Test No.	Chromosome	Fitness
1	81;55;40;34;21;16;	43.01792191
2	55;34;22;21;20;16;	43.18348209
3	55;34;21;16;15;10;	43.24261209
4	55;40;34;29;16;8;	43.19743927
5	218;36;34;22;16;8;	43.26475045

The fitness function value achieved in the executed tests differs mutually only a little, while from the comparison of the genes value of the individual chromosomes it can be seen that their dispersion is quite large. This proves that the genetic algorithm behavior does not depend only on the previously defined parameters for the individual experiments, but also on the arrangement of the candidates for the pilot buses in the candidate file. Regarding this information I can assume, that it is just the downward arrangement of candidates according to the value of the coefficient of sensitivity that has unfavorable influence on the tuning of the genetic algorithm parameters.

By the comparison of the individual genes value in 0 with the genes value in 0 we can learn that there are such genes in the chromosome, that can achieve the best fitness function value enumerated for one pilot bus (n =

1) via the global search method, which proves the suitability of implementing the method of regressive selection developed within 21.

The convergence speed of these tests to the suboptimal solutions is shown in the following graph (Fig. 19), which clearly illustrates the fast finding of the suboptimal solution (in the course of first 100 generations) with subsequent stagnation in this identified local extreme. However, there is an assumption that the achieved results can be improved by further tuning of the algorithm parameters.

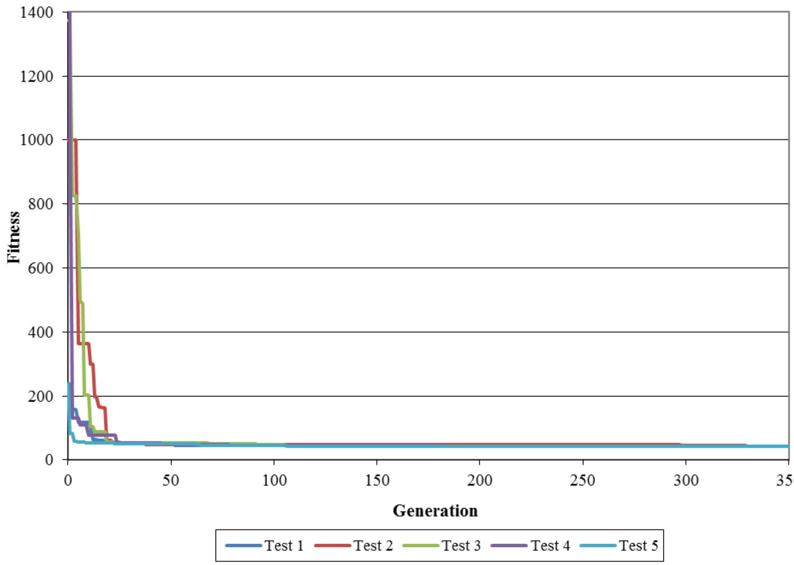


Fig. 19 *Convergence of calculation for n = 6*

The second executed experiment comprised of three tests. The parameters of the tests carried out within this experiment are as follows: number of generations = 800, size of population = 30, number of the best individuals = 3, number of original individuals = 3, size of work group = 24 (12 of the best ones 12, 12 of the original ones), probability of normal mutation = 0.2, probability of additive mutation = 0.2, amplitude of additive mutation = 5, and the number of points of random crossover = 3. The searched space was reduced by the method of the sensitivity threshold value coefficient of σ = 0, i.e. its size was 598 candidates, which corresponds with the value of 61 936 603 045 317 combinations. The number of the instances of GA Slave applications participating in the calculation = 8.

The results of these tests illustrated in the graph in Fig. 20 prove the findings of the previous experiment in terms of the algorithm convergence speed. It can be observed again that the algorithm converged to the suboptimal solutions in the course of the first 100 generations.

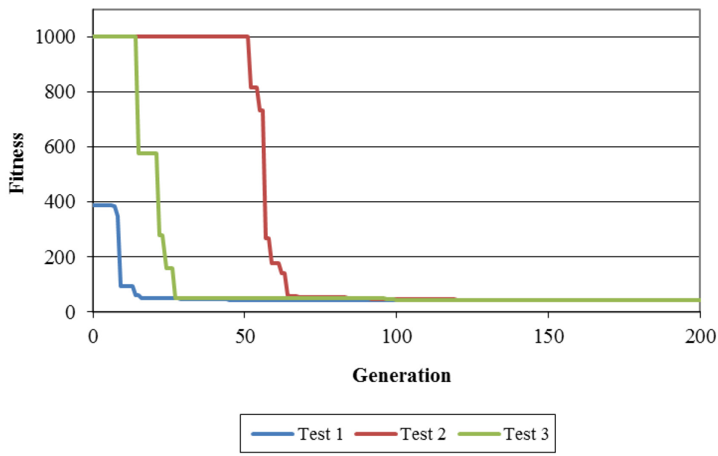


Fig. 20 *Convergence of calculation for n = 6 (accomplished by an error)*

5.4.5 Evaluation

The results of the experiments described in this Chapter prove the potential of genetic algorithms to solve the main task defined in the monograph. Despite the time consuming calculations considering the genetic algorithms essence it was shown that the designed architecture is profitable, since by using the parallel approach to evaluate the fitness function and by the suitable database use it was possible to considerably eliminate this harmful property. It is also obvious, that by further tuning of the genetic algorithm parameters, or by the suitable change of its structure it is possible to achieve better results.

5.5 Verification of parallel genetic algorithm functionality

The Chapter describes the experimental verification of the operation of the parallel genetic algorithm implemented in terms of the designed architecture. The migration structure of the best individuals of the subpopulation is shown in Fig. 21. The chromosomes migration occurs in random generations with a certain probability, which is freely adjustable for each instance of the GA Master application. When the requirement of probability is met, the instance administrating the A subpopulation request for the best chromosome from the last generation of the B subpopulation occurs. It operates this way also in the case of the instance administrating the B subpopulation. As the description shows, for the needs of the experiment two independent instances of the GA Master application were implemented.

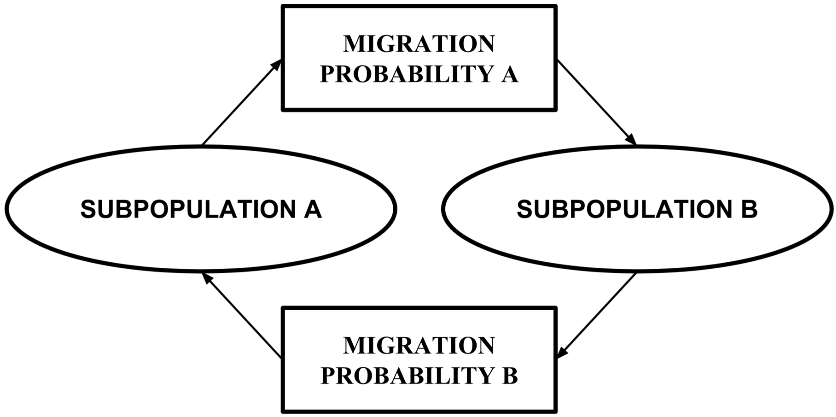


Fig. 21 *Structure of tested parallel genetic algorithm*

For the individual tests the following common parameters were set up: number of generations = 600, size of subpopulation A and B = 20, number of instances of the GA Master application = 2 and number of instances of the GA Slave application = 8. The searched space was reduced by the method of regressive selection on the basis of the findings mentioned in the previous Chapter, i.e. its size was 50 candidates which corresponds with the value of 19 600 combinations.

The parameters that were changed during the experiment for the individual tests are shown in the following table (Table 8).

PARAMETERS OF GENETIC ALGORITHM
FOR INDIVIDUAL TESTS

Table 8

Test No.	1		2		3		4		5	
Subpopulation	A	B	A	B	A	B	A	B	A	B
Number of best individuals	2	2	2	2	2	2	1	2	1	1
Number of original individuals	2	2	2	2	2	2	1	2	1	1
Size of operational group (best one + original)	16 (8+ 8)	16 (8+ 8)	16 (8+ 8)	16 (8+ 8)	16 (8+ 8)	16 (8+ 8)	18 (9+ 9)	16 (8+ 8)	18 (9+ 9)	18 (9+ 9)
Probability of normal mutation	0.2	0.6	0.2	0.4	0.2	0.4	0.2	0.4	0.1	0.9
Probability of additive mutation	0.2	0.6	0.2	0.5	0.2	0.5	0.2	0.5	0.1	0.9
Amplitude of additive mutation	1	5	1	8	1	8	1	8	1	8
Number of points of random crossover	1	1	1	1	1	1	1	1	1	1
Probability of migration	0.3	0.3	0.2	0.15	0.2	0.15	0.2	0.1	0.2	0.05

The convergence of the individual tests executed within the subject experiment is illustrated in several graphs for the best possible comparison. The graphs in Fig. 22 and Fig. 23 compare the convergence of subpopulations A and B for all tests carried out. The graphs in Fig. 24 to Fig. 28 show the common curve of the convergence of the subpopulations A and B separately for the individual tests.

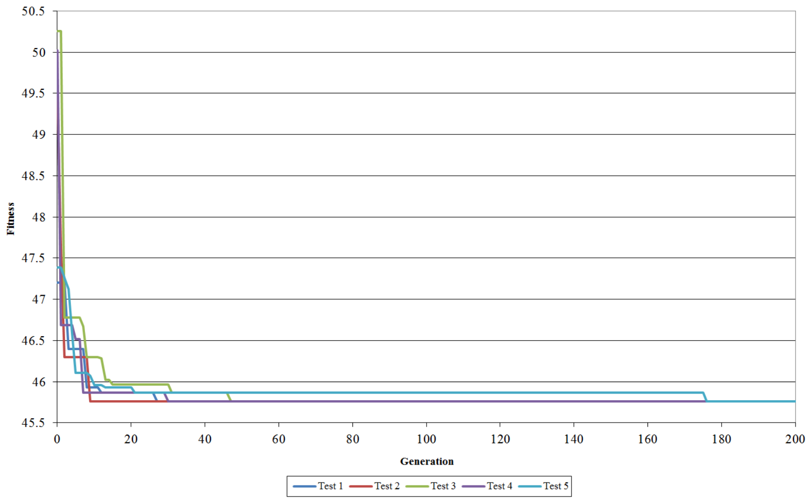


Fig. 22 *Convergence of subpopulation A for 5 setups*

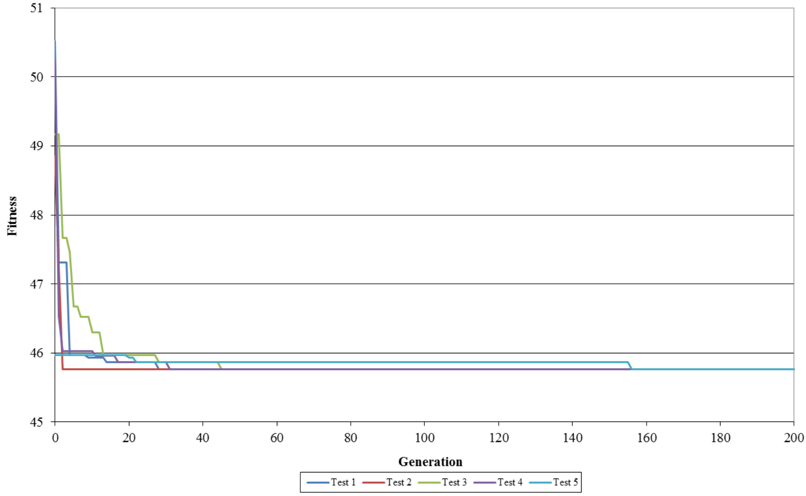


Fig. 23 *Convergence of subpopulation B for 5 setups*

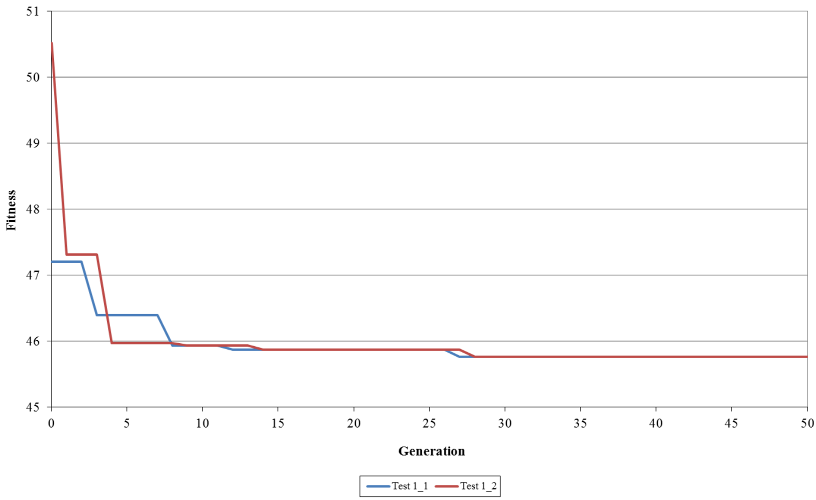


Fig. 24 *Convergence of subpopulations A and B for Test No. 1*

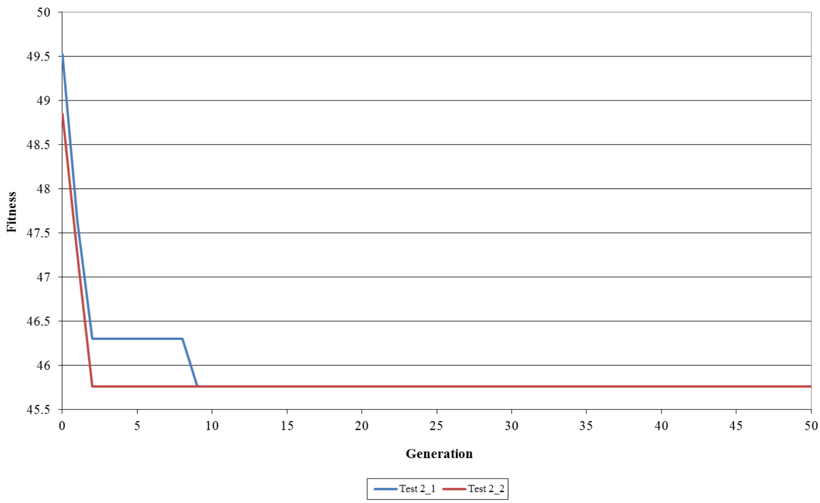


Fig. 25 *Convergence of subpopulations A and B for Test No. 2*

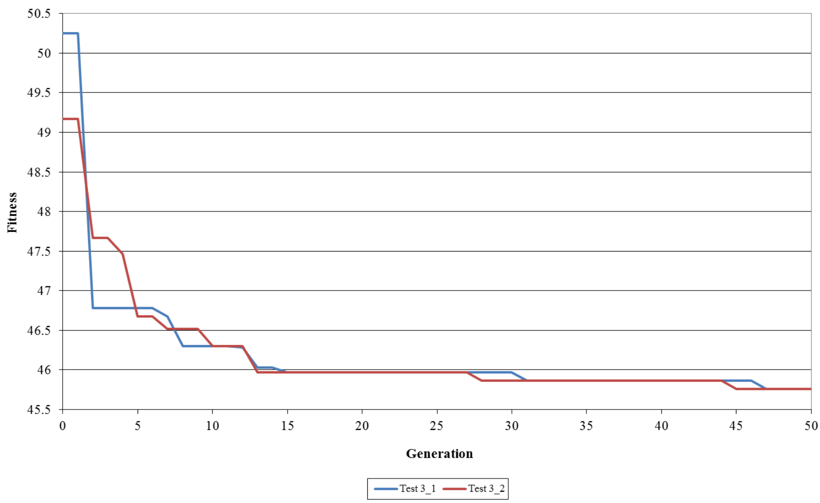


Fig. 26 *Convergence of subpopulations A and B for Test No. 3*

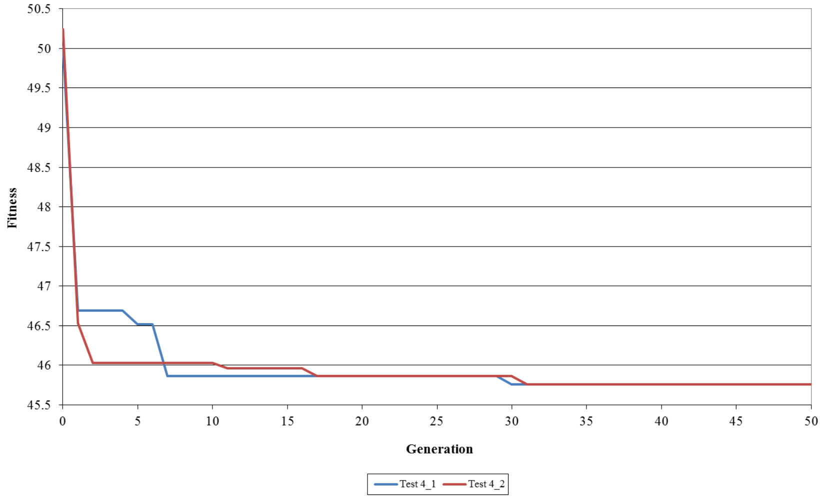


Fig. 27 *Convergence of subpopulations A and B for Test No. 4*

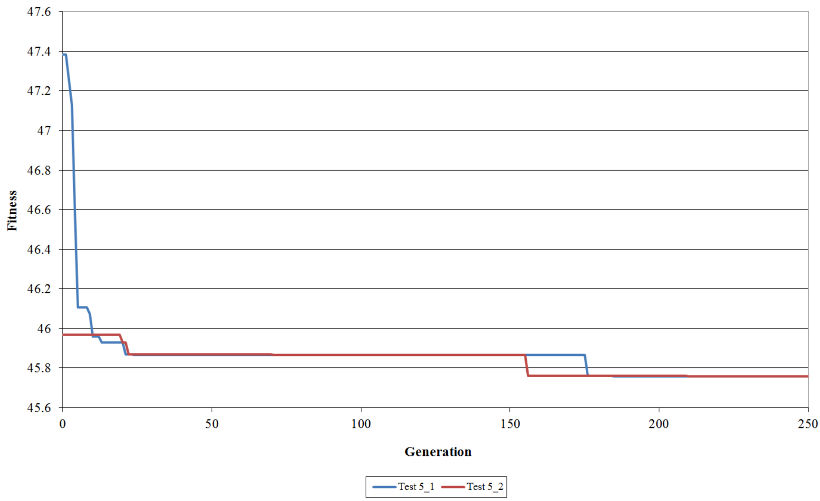


Fig. 28 *Convergence of subpopulations A and B for Test No. 5*

The individual graphs show the mutual influencing of the subpopulations. The influence of the parameters of the individual instances of the GA Master application, and hence the parameters of the individual genetic algorithms on the subpopulation development or on the convergence speed of the result to the global optimum can be also noticed. It can be best seen in the graph related to Test No. 5 Fig. 28. In this test the mutation parameters for the A subpopulation were set to relatively low values, which resulted in the obstruction of the algorithm in the small space and the long-term stagnation in the local extreme. On the contrary, the mutation parameters of B subpopulation were set to quite high values, which meant that the algorithm moved in a much larger space; however, the result was the same as in the case of A subpopulation, which means that it resulted again to the long-term stagnation in the local extreme. Nevertheless, the result finally converged to the global optimum although with the higher number of generations than in other cases as can be seen in the graphs.

These results unambiguously show that the parallel genetic algorithm has within the designed architecture its reason and can improve the properties of the ordinary genetic algorithm particularly due to the capability to search a much larger space in the same time. Its increased ability to escape the local extreme space is its undisputable advantage as well.

In the conclusion of the Chapter I would like to evaluate the average total time for the defined task calculation. The time will be evaluated for the processing of 600 generations; nevertheless, we have to realize that in the given configuration the system operated parallel with two subpopulations, which correspond with the number of 1200 generations processed within one calculation task. The calculations of computing time for the individual experiments are shown in Table 9.

AVERAGE COMPUTING TIMES FOR TESTS No. 1 TO 5 Table 9

Test No.	**Average time of one population evaluation [s]**	**Number of generations [pcs]**	**Average time consumption of the task**	
			[s]	**[h:m:s]**
1	7.99	600	4794	**1:19:55**
2	4.72	600	2832	**0:47:12**
3	4.86	600	2916	**0:48:36**
4	8.68	600	5208	**1:26:49**
5	6.18	600	3708	**1:01:48**

5.6 Description of the experimental system set

Experimental verification of the designed architecture required the formation of a small computer network to show its implementation suitability via the grid computing technology. For this purpose the capacities available within the specialized automation laboratory were utilized and subsequently further experiments were carried out via virtual PCs. These were actuated on virtualized servers which are available at the Faculty.

In both cases the network was formed by using 4 PCs equipped with MS Windows XP operational system. In case of real PCs, in the operational system several specialized applications for the needs of teaching were installed, in case of virtual PC they were the sets equipped with common office software.

The database server was implemented on a separate PC, in this case it is a common office set as well.

The first experiments were actuated so that the instance of the GA Master application, whose role is described in Chapter 4.4.2, was actuated together with the instances of the GA Slave application. Finally, the instances of the GA Master application were actuated on the PC with the database server.

The network structure used in the experiments execution is illustrated in Fig. 29. The database server and PC 1 to PC 4 communicate mutually together via LAN (Local Area Network) loaded by the common operational communication within the everyday life at the Faculty.

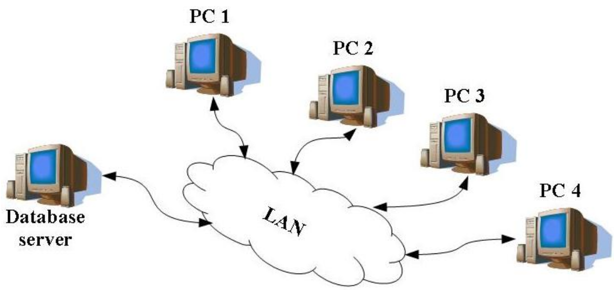


Fig. 29 *Network structure used for experiments execution*

During the majority of experiments two instances of the GA Slave application were set up on the sets of PC 1 to PC 4. In the case of experiments executed via virtual PC it was possible to simply monitor various parameters characterizing the load of the system. The most suitable information in this case is provided by the percentage of CPU load.

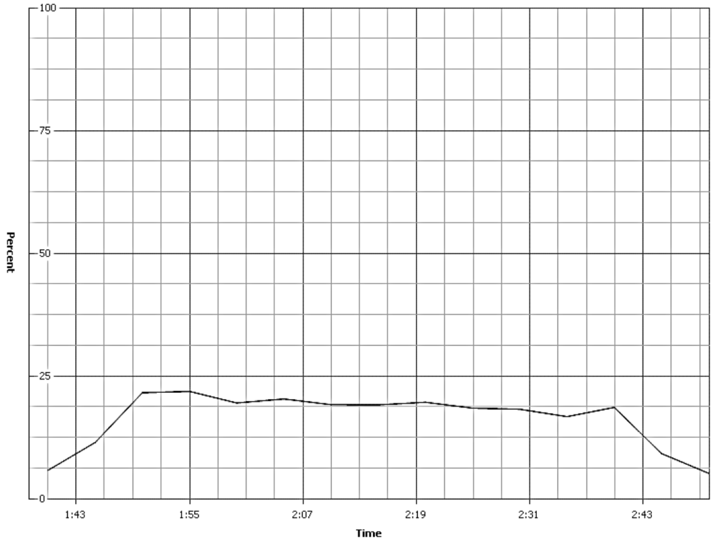


Fig. 30 *Exploitation of CPU of virtual PC 1*

As seen in the selected CPU load characteristic illustrated in Fig. 30, when two instances of the GA Slave application ran on it, the value of load did not exceed 25%. Regarding these results as well as the information in Chapter 4.2 I can assert, that it is meaningful to implement the designed architecture with the use of grid computing technology. Because it would be possible to run the instances of the GA Slave application on the background of the running operating system without the fact that the additive load occurred in the common activity of the user participating in the grid.

As the monitored CPU load characteristics for other used virtual PC were almost identical, their characteristics are mentioned in the annex (0).

5.7 Evaluation

The presented experiments unambiguously prove the assumptions which were considered in the design of the system architecture of the optimal pilot buses selection for the needs of the power system dispatch control in real time described in Chapter 4.4. It is necessary to realize that the implementation of the subject architecture carried out within the monograph cannot be considered definite in any case. Its purpose was purely experimental and it had to provide the verification of the designed architecture.

In the course of experiments several constraints in the implementation of the tested system were identified, e.g.:

- *requirements for the database server* – for the needs of experiments the system of database control was setup for the common office set, in this case some difficulties with the lack of system means could occur, therefore, it is necessary to consider a dedicated database server for the next implementation ;
- *method of communication* – in the communication the periodical information processing was utilized, the period was small (regularly milliseconds), nevertheless, despite this fact it was possible to notice its influence on the computing time, which was neglected in the result;
- *simulation software* – regarding the nature and properties of the MODES system, it would be suitable to find suitable simulation software which could be implemented into the application without any complications.

CONCLUSION

The scientific monograph submitted deals with the issue of optimal pilot buses selection of the dispatch voltage buses control of the complex power system in real time. As aforementioned several times, the selection of pilot buses represents a multi-dimensional optimizing issue, since depending on the size of the related power system and the dimension of the *n-ple* of searched pilot buses the state space grows combinatorially. The monograph is based on the methods for the reduction of the number of candidates for pilot buses and on the methods of optimal pilot buses selection, prevailingly developed in 21 and 31. These methods are modified and the original system architecture of the optimal pilot buses selection for the dispatch voltage buses control of the complex power system is designed. The related architecture considers the principles of the parallel approach to solving the complex calculation tasks as well as the system implementation possibilities in the environment of the grid computing structures.

Implementation of the prototype software based on the designed original architecture served for the experimental verification of the assumptions is articulated in Chapter 4.4, while the experiments unambiguously proved the potential of the designed original architecture.

Perspectives of further development

Considering the results and analyses achieved in the experiments, we can provide the perspectives of further research in the field of pilot buses selection:

- Optimization of structure and parameters of basic and subsequent parallel genetic algorithm used within the designed original architecture, since the experiment results show that even small changes of some parameters significantly influence the abilities of the used genetic algorithms.
- Incorporation of the expert system operating on the principle of the electric distance directly into the structure of used genetic algorithms.
- Utilization of artificial intelligence in searching for analogic situations in the power system in the given time moment.
- Orientation of research in Smart Grid technology.

REFERENCES

1. BABJAK, J. Genetické algoritmy (*Genetic algorithms*), Computer world 06/2003. [online, 25 Jul 2008] URL: <http://archiv.computerworld.cz/>
2. BELÁŇ, A., SEKAJ, I., VALO, J. 2004. Optimalizácia napätí v elektrizačnej sústave pomocou genetických algoritmov *(Optimization of power system voltage via genetic algorithms)*. In: *6th International Conference Control of Power Systems*. Štrbské Pleso.
3. BLANCHON, G. 1972. A New Aspect of Studies of Reactive Energy and Voltage of the Networks. In: *Proceedings of 1972 Power Systems Computation Conference (PSCC)*. Grenoble (France).
4. Čo je to grid computing a čo je to vlastne BOINC? (*What is grid computing and what is BOINC?*) [online, 16 Apr 2009], URL: <http://boinc.sk/boinc>.
5. CANTÚ-PAZ, E. *A Survey of parallel genetic algorithms.* [online, 19.6.2011], URL: <http://neo.lcc.uma.es/cEA-web/documents/cant98.pdf>.
6. CAÑIZARES, A. C., CAVALLO, C., POZZI, M., CORSI, S. 2005. Comparing secondary voltage regulation and shunt compensation for improving voltage stability and transfer capability in the Italian power system. In: *Electric Power System Research*, 73/2005, pp. 67-76.
7. CONEJO, A., AGUILAR, M. J. 1996. A nonlinear approach to the selection of pilot buses for secondary voltage control. In. *Proceedings Power System Control and Management, 16-18 Apr 1996*, Conference Publication No. 421. London : IEEE, pp. 191-195.
8. CONEJO, A., AGUILAR, M. J. 1998. Secondary voltage control: Nonlinear selection of pilot buses, design of an optimal control law, and simulation results. In: *IEE Proc.- Gener. Transm. Distrib.*, 145, pp. 77-81.
9. CONEJO, A., FUENTE, J. I., GÖRANSSON, S. 1994. Comparison of alternative algorithms to select pilot buses for secondary voltage control in electric power networks. In: *7th Mediterranean Electrotechnical Conference*, pp. 940 – 943.
10. CONEJO, A., GÓMEZ, T., FUENTE, J. I. 1993. Pilot-bus selection for secondary voltage control. In: *ETEP* 3, pp. 359-366.

11. DEVARAJ, D., YEGNANARAYANA, B. 2005. Genetic-algorithm-based optimal power flow for security enhancement. In: *IEE Proceedings – Generation, Transmission and Distribution*, Volume **152(**6), pp 899 – 905.
12. Diskrétna matematika 2. *(Discrete mathemtics 2)* [online, 16 Jul 2008], URL: <http://januska.unitra.sk/kombinT1.htm>.
13. FOLTIN, M., ERNEK, M. 2007. Model of Slovak Power System Using Simpowersystem. In: *Technical Computing Prague 2007: 15th Annual Conference Proceedings*. Prague, Czech Republic, 14.11.2007. - Praha: Humusoft s.r.o. ISBN 978-80-7080-658-6 - CD-Rom.
14. FOSTER, I., KESSELMAN, C. 2004. *The Grid: Blueprint for a New Computing Infrastructure II*. Morgan Kaufmann, pp. 35-64.
15. Genetický algoritmus. *(Genetic algorithm)* [online, 15.8.2008]. URL: <http://sk.wikipedia.org/wiki/Genetick%C3%BD_algoritmus>
16. GÓMEZ, T., CONEJO, A. J., FUENTE, J. I., PAGOLA, F. L., REHN, C. J. 1992. Decentralized Secondary Voltage Control and Pilot Bus Selection. In: *IFAC Symposium on Control of Power Plants and Power Systems*. Germany: Munich.
17. GRIGER, V. Elektrizačná sústava a jej regulácia. *(Power system and its regulation)* [online, cit. 21 Aug 2008], http://www.seas.sk/_cms_/_files/812/t020801.htm.
18. ILIC-SPONG, M., CHRISTENSEN, J., EICHORN, K. L. 1988. Secondary voltage control using pilot point information. In: *IEEE Transactions on Power Systems*, 3, pp. 660 – 668. ISSN 0885-8950
19. KAJAN, S., SEKAJ, I., MRAFKO, L. 2007. Parameter Tuning of a Stream Pressure Controller Using Genetic Algorithm and Neural Model. In: *Process Control 2007 : 16th International Conference.* Bratislava: ChtF STU. ISBN 978-80-227-2677-1 - CD-Rom.
20. KOLCUN, M., GRIGER, V. 2003. *Riadenie prevádzky elektrizačných sústav (Operation control of power systems)*. Košice: Mercury – Smekal. ISBN 80-89061-76-1
21. KUNÍK, S. 2008. *Optimálny výber pilotných uzlov v sekundárnej regulácii zložitej ES (Optimal selection of pilot bus in secondary regulation of complex ES)* Doktorandská dizertačná práca *(Doctoral Dissertation thesis)*. Trnava: KAIA MtF STU, 116 p. Supervisor: Dušan Mudrončík.
22. KVASNIČKA, V., POSPÍCHAL, J., TIŇO, P. 2000. *Evolučné algoritmy (Evolution algorithms)*. Bratislava: STU. ISBN 80-227-1377-5

23. LAGONOTTE, P., SABONNADIERE, J. C., LEOST, J.Y., PAUL, J. P. 1989. Structural analysis of the electrical system: Application tosecondary voltage control in France. In: *IEEE Transactions on Power Systems*, 2, pp. 479-486.
24. MÁSLO, K., VÁGNER, P. Síťový simulátor MODES *(MODES network simulator)* [online, 12 Aug 2008], <http://modesinfo.com/>
25. MÁSLO, K. Popis modelování přechodných dějů v ES programem MODES v.2.2./8. *(Description of temporary actions modelling in ES via MODES v. 2.2./8 program).* Používateľská príručka MODES 2.2.8. 2003 *(MODES v. 2.2./8 program user manual)* .
26. MÁSLO, K. 1995. *Průvodce použití programu MODES při analýze dynamického chování elektrizační soustavy. Metodický průvodce programu MODES (Guide to MODES program use in the analysis of ES dynamic behavior).* 2. vydání *(2nd edition*).
27. MÁSLO, K. 2003. *Uživatelská příručka programu MODES 2.2/8 (User guide to MODES 2.2.8). Používateľská príručka k MODES 2.2.8. 2003.*
28. MLEJNEK, D. 1999. *Výber pilotných uzlov v elektrizačnej sústave (Pilot buses selection in power system). Diplomová práca (Master thesis).* Bratislava: FEI STU, 77 p. Supervisor: Dušan Mudrončík
29. MUDRONČÍK, D., KOZÁKOVÁ, A., MLEJNEK, D. 2000. Analysis of a power system under secondary voltage control. In *4th International Conference Control of Power Systems.* Bratislava: FEI Bratislava, pp. 341-345.
30. PAUL, J. P., LEEOST, J. M. 1987. Survey of the Secondary Voltage Control in France: Present Realization and Investigations. In: *IEEE Transactions on Power Systems*, 2, pp. 505-511.
31. RAKOVSKÝ, M. 2006. Redukcia výberu pilotných uzlov v sekundárnej regulácii *(Reduction of pilot buses selection in secondary control).* Doktorandská dizertačná práca *(Doctoral Dissertation thesis)*, Trnava: KAIA MtF STU. 101 p. Supervisor: Dušan Mudrončík
32. SEKAJ, I., FOLTIN, M. 2003. Matlab Toolbox – Genetické algoritmy *(Genetic algorithms)*. In: *MATLAB 2003: Sborník příspěvků 11. ročníku konference. Díl I (Conference proceedings. Volume I).* Praha: ČVUT, pp. 514 -519.
33. SEKAJ, I. 2005. Evolučné výpočty – a ich využitie v praxi *(Evolution calculations – and their utilization in practice).* Bratislava: IRIS. ISBN 80-89018-87-4

34. SEKAJ, I. 2001. Genetické algoritmy (1) – čo (a načo) sú genetické algoritmy. (Genetic algorithms (1) – what are they and what for). In: *AT&P Journal*, 11, pp. 46-48.
35. SEKAJ, I. 2001. *Genetické algoritmy (2) – použitie genetických algoritmov na riešenie optimalizačných úloh (Genetic algorithms (2) – use of genetic algorithms for optimization tasks investigation).* In *AT&P Journal*, 12, s. 39-40.
36. SEKAJ, I. 2002. *Toolbox – genetické algoritmy. Používateľská príručka (Toolbox – genetic algorithms. User guide). Interný material (Internal material)*. Bratislava: STU v Bratislave FEI.
37. SCHREIBER, P., MORAVČÍK, O., VAŽAN, P. 2003. Ovplyvňovanie chovania genetických algoritmov vstupnými parametrami (Influencing the genetic algorithms behavior via input parameters). In: *Materials Science and Technology, vol. 2.* [online]. [cit. 10 Jul 2003]. ISSN 1335-9053
38. STANKOVIC, A., ILIC-SPONG, M., MARATUKULAM, D. 1991. Recent results in secondary voltage control of power systems. In: *IEEE Transactions on Power Systems* 6, pp. 94-101. ISSN 0885-8950
39. THORP, J. S., ILIC-SPONG, M., VARGHESE, M. 1986. An optimal secondary voltage-VAR control technique. In: *Automatica*. Journal of IFAC, 22, pp. 217 – 222. ISSN 0005-1098
40. UZUNOVIC, E., CAÑIZARES, A. C., REEVE, J. 1997. Fundamental Frequency Model of Static Synchronous Compensator. In: *Proceedings of the North American Power Symposium (NAPS)*, Wyoming: Laramie, pp. 49-54.
41. VIALAS, C. – PAUL, J. P. 1988. Trends in automatic regional voltage control of the French EHV power system: the effect on communication requirements. In: *Power Systems-Modelling and Control Applications*. *IFAC symposium*. Brussels: pp. 161-165.

LIST OF AUTHOR´S PUBLICATIONS

1. BEZÁK, T., ELIÁŠ, A., KOPČEK, M., STRÉMY, M. 2008. Renovácia riadiaceho systému kontúrovacej píly na rezanie expandovaného polystyrénu (Renovation of control system of contour cutter for cutting of expanded polystyrene). In: *Materials Science and Technology,* 8. [online]. ISSN 1335-9053

2. KOPČEK, M., MUDRONČÍK, D., KUNÍK, S. 2008. Implementácia dynamického nelineárneho modelu viacstrojovej elektrizačnej sústavy (Implementation of dynamic nonlinear model of multimachine power system). In: *Materials Science and Technology, 7.* [online]. ISSN 1335-9053
3. KOPČEK, M., ELIÁŠ, A. 2008. Realizácia SCADA systému ako podporného nástroja pre návrh metodiky zisťovania zostatkového výkonu vyhoreného jadrového paliva. (SCADA system development as a supportive tool for the measuring of residual power of burned - out nuclear fuel.). In: *Materials Science and Technology, 8* [online]. ISSN 1335-9053
4. KUNÍK, S., KOPČEK, M., MUDRONČÍK, D. 2008. Dostupné metódy výberu pilotných uzlov pre sekundárnu reguláciu elektrizačnej sústavy (Pilot bus selection available methods for the SVC of a power system). In: *Materials Science and Technology, 6.* [online]. ISSN 1335-9053
5. KUNÍK, S., MUDRONČÍK, D., KOPČEK, M. 2008. Nové prístupy k výberu pilotných uzlov pre sekundárnu reguláciu ES (New algorithms for pilot bus selection in SVC of a power system). In: *Materials Science and Technology*, 7. [online]. ISSN 1335-9053
6. KUNÍK, S., MUDRONČÍK, D., KOPČEK, M. 2008. Prehľad zariadení FACTS z pohľadu sekundárnej regulácie ES (Overview of the most commonly used facts controlers in SVC of a power system). In: *Materials Science and Technology, 7* [online]. ISSN 1335-9053
7. STRÉMY, M., ELIÁŠ, A., KOPČEK, M., BEZÁK, T., ZÁVACKÝ, P. 2009. Residual power determination of the fissionable fuel. In: *Annals of DAAAM and Proceedings of DAAAM Symposium* 2009, **20**(1). ISSN 1726-9679. Annals of DAAAM for 2009 & Proceedings of the 20th international DAAAM symposium "Intelligent manufacturing & automation: Focus on theory, practice and education" Vienna: DAAAM International Vienna, pp. 0297-0298. ISBN 978-3-901509-70-4
8. TANUŠKA, P., KUNÍK, S., KOPČEK, M. 2009. *Exothermic CSTR:* Modeling, control & simulation. In: *Annals of DAAAM and Proceedings of DAAAM Symposium*, **20**(1). ISSN 1726-9679. Annals of DAAAM for 2009 & Proceedings of the 20th international DAAAM symposium "Intelligent manufacturing & automation: Focus on theory, practice and education" Vienna: DAAAM International Vienna, pp. 0203-0204. ISBN 978-3-901509-70-4

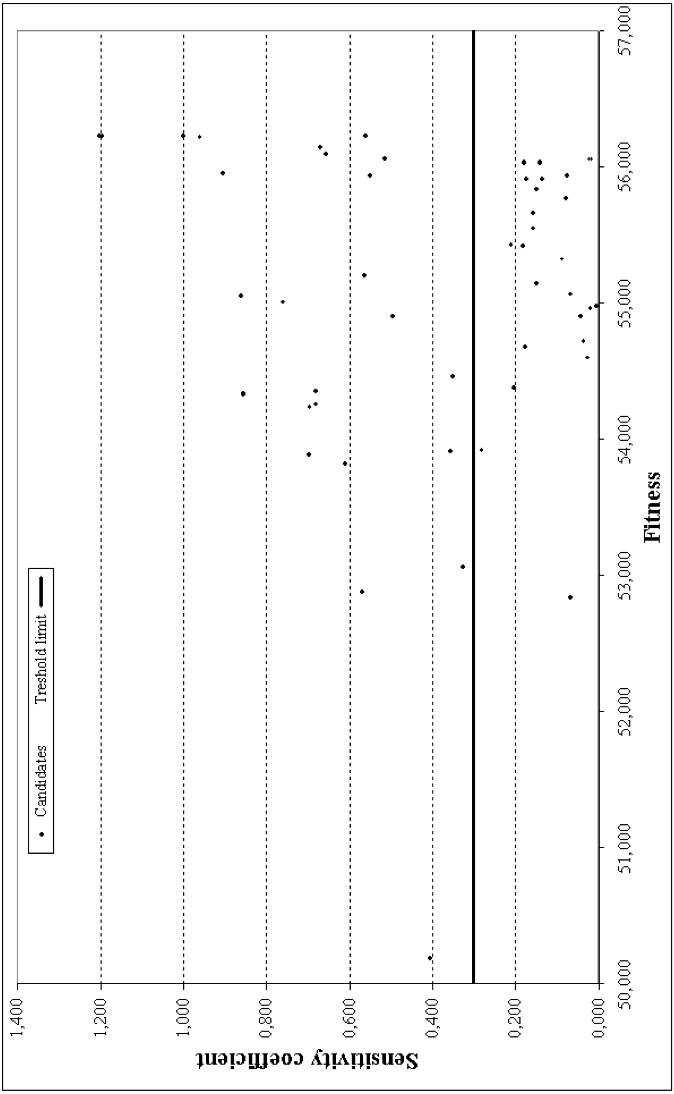
9. KOPČEK, M., KUNÍK, S. 2008. Dynamic model of multi - machine power system: the mathematical background. In XXXIII. Seminar ASR ´2008. Instruments and Control: Ostrava: Fakulta elektrotechniky a informatiky VŠB. ISBN 978-80-248-1727-9
10. KOPČEK, M. - KUNÍK, S. - MUDRONČÍK, D. 2008. Dynamic model of multi-machine power system. In: *Process Control 2008 : Proceedings of the 8th International Scientific-Technical Conference.* Pardubice: University of Pardubice. ISBN 978-80-7395-077-4
11. KOPČEK, M., BEZÁK, T., ZÁVACKÝ, P., FRIDRICH, V. 2010. Inspection stand for monitoring of spent nuclear fuel SVYP-440. In: *Process Control 2010 : 9th International Conference.* Pardubice: University of Pardubice, C060a-1-5. ISBN 978-80-7399-951-3
12. KOPČEK, M., MUDRONČÍK, D. 2010. Optimal pilot bus selection using GA and computation parallelism. In: *Process Control 2010 : 9th International Conference.* Pardubice : University of Pardubice, C060b-1-4. ISBN 978-80-7399-951-3
13. KUNÍK, S., MUDRONČÍK, D., KOPČEK, M. 2008. Optimal pilot bus selection using GA. In: *Process Control 2008 : Proceedings of the 8th International Scientific-Technical Conference.* Pardubice: University of Pardubice. ISBN 978-80-7395-077-4
14. KUNÍK, S., KOPČEK, M. 2008. Optimal pilot bus selection using GA Delphi unit. In: *XXXIII. Seminar ASR ´2008. Instruments and Control.* Ostrava: Fakulta elektrotechniky a informatiky VŠB. ISBN 978-80-248-1727-9
15. TANUŠKA, P., VERSCHELDE, W., KOPČEK, M. 2008. Proposal of a Data Warehouse Test Scenario. - 1/4078/07. In: *Ecumict 2008 : Proceedings of the Third European Conference on the Use of Modern Information and Communication Technologies.* Nevelland v.z.w., pp. 403-409. ISBN 9-78908082-553-6
16. TANUŠKA, P., KOPČEK, M. 2007. The technological process control through the use of virtual controllers. In: *Mašinostrojenije i technosfera XXI veka : Sbornik trudov XIV meždunarodnoj naučno-techničeskoj konferenciji.* DonNTU, pp. 259-263, Tom 5. ISBN 966-7907-22-8
17. KOPČEK, M., MUDRONČÍK, D., KUNÍK, S., CIBULÁŠ, V. 2007. Automated system of temperature heating and cooling control of resistive furnace. In: *Process Control 2007 : 16th International Conference*. Bratislava: STU in Bratislava ChtF, 063.pdf. ISBN 978-80-227-2677-1

18. KOPČEK, M., KUNÍK, S. 2008. Mathematical background for a dynamic model of multi - machine power system. In*: International Doctoral Seminar 2008 : Proceedings.* Trnava: AlumniPress, pp. 148-151. ISBN 978-80-8096-058-2
19. KOPČEK, M., ELIÁŠ, A., BEZÁK, T. 2009. Scada for the measuring of burned - out nuclear fuel residual power. In*: International Doctoral Seminar 2009 : Proceedings.* Trnava: AlumniPress, pp. 148-153. ISBN 978-80-8096-088-9
20. KUNÍK, S., KOPČEK, M. 2008. GA delphi unit for pilot bus selection. In: *International Doctoral Seminar 2008 : Proceedings*. Trnava: AlumniPress, pp. 166-170. ISBN 978-80-8096-058-2
21. KUNÍK, S., MUDRONČÍK, D., KOPČEK, M. 2007. Optimum selection of pilot buses in secondary voltage control. In: *Process Control 2007 : 16th International Conference.* Bratislava: STU in Bratislava ChtF, 028.pdf. ISBN 978-80-227-2677-1
22. KUNÍK, S., MUDRONČÍK, D., KOPČEK, M., STRÉMY, M. 2007. Virtual and analogue model of a continuous stirred tank reactor. - článok je uverejnený v časopise: AT&P Journal Plus, recenzované periodikum vedeckých a inžinierskych publikácií *(paper published in AT&P Journal Plus, reviewed periodical of scientific and engineering publications)*. ISSN 1336-5010, No..2 (2007), pp. 24-27, (authors: Kuník Stanislav - Mudrončík Dušan - Kopček Michal). In: Process Control 2007 : 16th International Conference. Štrbské Pleso, Slovak Republic, 11.-14.6.2007. - Bratislava: STU v Bratislave ChtF, 010.pdf. ISBN 978-80-227-2677-1
23. STRÉMY M., KOPČEK, M., JEDLIČKA, M., ŠKULAVÍK, T., BEZÁK, T. 2011. *Úvod do programovateľných logických automatov (Introduction to programmable logic automatic machines)*. Trnava: Qintec s.r.o., 172 p. ISBN 978-80-969846-9-5
24. KOPČEK, M., MUDRONČÍK, D. 2011. Using GA and computation parallelism for the optimal pilot bus selection. In: *Lecture Notes in Electrical Engineering*, Vol. 122. ISSN 1876-1100. Advances in Automation and Robotics, Vol. 1: In: *International Conference on Automation and Robotics, ICAR 2011*. Berlin: Springer-Verlag Berlin Heidelberg, pp. 323-328. ISBN 978-3-642-25552-6

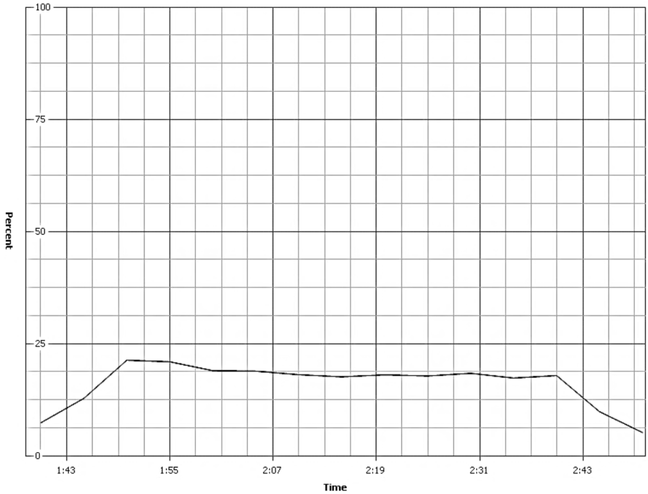
25. KOPČEK, M. 2011. *Optimalizácia procesov elektrizačnej sústavy na primárnej a sekundárnej úrovni riadenia (Optimization of power systems processes on primary and secondary control levels).* Dissertation Thesis. Trnava: STU Bratislava MTF UIAM, 114 p.
26. STRÉMY M., KOPČEK, M., BEZÁK, T. 2011. Introduction to combined dynamic systems. In: *Annals of Faculty of Engineering Hunedoara - Journal of Engineering*. - Tom IX, Fasc. 3, pp. 323-326. ISSN 1584-2673
27. STRÉMY M., KOPČEK, M. 2011. The energy spectrum of the combined systems. In: *Annals of Faculty of Engineering Hunedoara - Journal of Engineering*. Tom IX, Fasc. 3, pp. 433-435. ISSN 1584-2673
28. KOPČEK, M., STRAŠIFTÁK, A., STRÉMY M. 2012. Možnosti spojenia stavebnice LEGO MINDSTORMS NXT a prostredí MATLAB / Simulink. In: *Journal of Information Technologies*, č. 1, p. 7-15, ISSN 1337-7469
29. BEZÁK, T., KUSÝ, M., KOPČEK, M., ELIÁŠ, M. 2012. Computational processing of experimental data from laser scanning confocal microscopy. In: *Transfer 2012: 13. medzinárodná vedecká konferencia*. Trenčín: Trenčianska univerzita Alexandra Dubčeka v Trenčíne. ISBN 978-80-8075-560-7
30. GESE, A., KOPČEK, M., MESZÁROS, A. 2012. Signal filtering for a nuclear reactor restarting. In: *IEEE Proceedings of the 2012 6th International IEEE Conference Intelligent Systems*, IEEE, pp. 363-367. ISBN 978-1-4673-2277-5
31. KOPČEK, M., ŠKULAVÍK, T. 2012. Experimental verification of the computational system for the optimal pilot bus selection. In: *Lecture Notes in Engineering and Computer Science,* WCECS 2012. Vol. II. ISSN 2078-0958 : World Congress on Engineering and Computer Science 2012. Proceedings IAENG & IET. Hong Kong : International Association of Engineers, pp. 1187-119. ISBN 978-988-19252-4-4
32. BEZÁK, T., KUSÝ, M., ELIÁŠ, M., KOPČEK, M. 2013. Surface roughness determination using laser scanning confocal microscope Zeiss LSM 700. ITMS: 26220120014. In: *METAL 2013: 22nd International Conference on Metallurgy and Materials*. Ostrava: TANGER s r.o. ISBN 978-80-87294-39-0

33. ŠKULAVÍK, T., KOPČEK, M., KOPČEKOVÁ, A. 2013. Fuzzy Control of Robotic Arm Implemented in PLC. In: *ICCC 2013: IEEE 9th International Conference on Computational Cybernetics. Proceedings.* Piscataway: IEEE, pp. 45-49. ISBN 978-1-4799-0060-2
34. KOPČEK, M., BEZÁK, T., ELIÁŠ, M., KUSÝ, M. 2013. Computer processing of data obtained using laser scanning confocal microscopy. - ITMS 26220120014. In: *ERIN 2013. Proceedings of abstracts with full papers on CD: 7th international conference for young researchers and Ph.D. students.* Bratislava: Nakladateľstvo STU, 2013. ISBN 978-80-227-3934-4
35. KOPČEKOVÁ, A., KOPČEK, M., ŠKULAVÍK, T. 2013. Virtual model of an electro-hydraulic system. In: *Proceedings of the 2013 International Conference on Process Control:* Piscataway: IEEE, p. 274-279. ISBN 978-80-227-3951-1

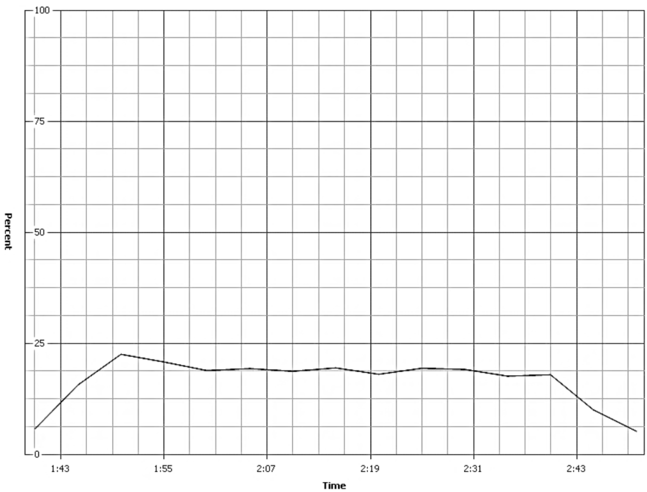
Annex A Graph complementing Table 5



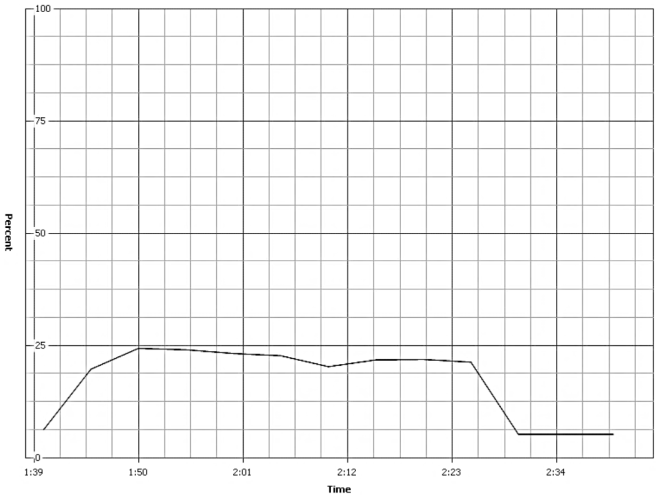
Annex B Characteristics of CPU load



Exploitation of virtual PC 2 CPU



Exploitation of virtual PC 3 CPU



Exploitation of virtual PC 4 CPU

CONTENTS

Author:	Ing. Michal Kopček, PhD.
Title:	OPTIMAL PILOT BUS SELECTION FOR THE SECONDARY VOLTAGE CONTROL USING PARALLELISM
Published in:	Ilmenau (Germany)
Edited by:	Universitätsverlag Ilmenau
1st edition:	2013
Editions:	Scientific monographs in Automation and Computer Science
Pages:	147 (figures 30; tables 9; annex 2; proofreading – Paul Wooliscroft)

Quires (20 standard pages): 5,89